NONPARAMETRIC DISTRIBUTED LEARNING ARCHITECTURE: ALGORITHM AND APPLICATION

By Scott Bruce, Zeda Li, Hsiang-Chieh Yang, and Subhadeep Mukhopadhyay*

Department of Statistics, Temple University

Abstract The big data era is here but where are the tools to analyze them? Dramatic increases in the size of datasets have made traditional "centralized" statistical inference techniques prohibitive. Surprisingly very little attention has been given to developing inferential algorithms for data whose volume exceeds the capacity of a single-machine system. Indeed, the topic of big data statistical inference is very much in its nascent stage of development. A question of immediate concern is how can we design a data-intensive statistical inference architecture without changing the basic fundamental data modeling principles that were developed for 'small' data over the last century? To address this problem we present `MetaLP`–a flexible and distributed statistical modeling paradigm suitable for large-scale data analysis where statistical inference meets big data technology. This generic statistical approach addresses two main challenges of large datasets: (1) massive volume and (2) variety or mixed data problem. We apply this general theory in the context of a nonparametric two sample inference algorithm for Expedia personalized hotel recommendation engine based on 10 million records of search results. Furthermore, we show how this broad statistical learning scheme (`MetaLP`) can be successfully adapted for 'small' data in resolving the challenging problems of Simpson's paradox and Stein's paradox. The R-scripts for MetaLP-based parallel processing of massive data by integrating with the Hadoop's MapReduce framework are available as supplementary materials.

1. Introduction. *Motivation.* Expedia (`www.expedia.com`) is the world's largest online travel agency. It has approximately 150 sites that operate in 70 countries around the world, with 50 million visitors a month and 200 mobile app downloads a minute. Expedia released a dataset (through the 2013 International Conference on Data Mining data competition) containing 52 variables of user and hotel characteristics (e.g. click-through data, hotel characteristics, user's aggregate purchase history, competitor price information) from over 10 million hotel search results collected over a window of the year 2013. The Expedia digital marketing team intends to understand: what visitors want to see in their Expedia.com search results, or in other words, what features of "search result impressions" lead to purchase with a goal of increasing user engagement, travel experience, and the conversion or booking rate. These factors will ultimately be used to predict the needs of consumers–to personalize, target, and provide consumers with content that is contextually relevant. For this purpose we

*Corresponding author. Scott Bruce, Zeda Li, and Hsiang-Chieh Yang are doctoral students. All authors contributed equally to this work.

Keywords and phrases: Distributed statistical learning, Big data inference, Nonparametric mixed data modeling, LP transformation, Meta-analysis, Confidence distribution.

develop a scalable distributed algorithm that can mine search data from millions of travelers in a completely nonparametric manner to find the important features that best predict customers' likelihood to book a hotel–an important large-scale machine learning problem, which is the main focus of this paper.

The Volume Problem. This kind of 'tall' data structure, whose number of observations can run into the millions and billions frequently arises in astronomy, marketing, neuroscience, e-commerce, and social networks. These massive datasets, *which cannot be stored or analyzed by a single computer 'all-at-once' using standard data analysis software*, create a major bottleneck for statistical modeling and inference. There is currently no efficient and flexible statistical inference model available to address this problem. We seek to develop a comprehensive framework that will allow data scientists to *systematically apply the tools and algorithms developed prior to the 'age of big data' for massive data problems* - thus filling a significant gap in current practice.

The Variety Problem. Another challenge is how to tackle the mixed data problem (Parzen and Mukhopadhyay, 2013) - one of the biggest unsolved problems of data science. The Expedia dataset contains variables of different types (e.g. continuous, categorical, discrete, etc.). The traditional statistical modeling approach develops tools that are specific for each data type. A few examples of traditional statistical measures for $(Y; X)$ data include: (1) Pearson's ϕ-coefficient: Y and X both binary, (2) Wilcoxon Statistic: Y binary and X continuous, (3) Kruskal-Wallis Statistic: Y discrete multinomial and X continuous, and many more. Computational implementation of traditional statistical algorithms for heterogeneous large datasets (like the Expedia search data) thus become dauntingly complex as they require the data-type information for each pair from the user to calculate the proper statistic. To streamline this whole process we need to develop unified computing algorithms with automatic data-driven adjustments that yield appropriate statistical measures without demanding the data type information from the user. We call this new computing culture United Statistical Algorithms (Parzen and Mukhopadhyay, 2013). To achieve this goal we design a customized discrete orthonormal polynomial-based transformation, the LP-Transformation, (Mukhopadhyay and Parzen, 2014) for any arbitrary random variable X, which can be viewed as a nonparametric data-adaptive generalization of Norbert Wiener's Hermite polynomial chaos-type representation (Wiener, 1938). This easy-to-implement LP-transformation based approach allows us to simultaneously extend and integrate classical and modern statistical methods for nonparametric feature selection, thus providing the foundation to build *increasingly automatic algorithms* for large complex datasets.

Scalability Issue. Finally the most crucial issue is to develop a scalable algorithm for large datasets like the Expedia example. With the evolution of big data structures, new processing capabilities relying on distributed, parallel processing have been developed for efficient data manipulation and analysis. Our technique addresses the question of how to develop a statistical inference framework for massive data that can fully exploit the power of parallel computing architecture and can be easily embedded into the MapReduce framework. We especially design the statistical 'map' function and 'reduce' function for massive data variable selection by integrating many modern statistical concepts and ideas introduced in Sections 2 and 3. Doing so allows for faster processing of big datasets while maintaining the ability to

obtain accurate statistical inference without losing information - thus providing an effective and efficient strategy for big data analytics. The other appealing aspect of our distributed statistical modeling strategy is that it is equally applicable for small and big data. Our modeling approach is generic and unifies small and big data variable selection strategies.

Organization. Motivated by the challenges of the Expedia case study, we design `MetaLP` to support fast statistical inference on very large datasets that provides scalability, parallelism, and automation. The article is organized as follows. Section 2 provides the basic statistical formulation and overview of the `MetaLP` algorithmic framework. Section 3 covers the individual elements of the distributed statistical learning framework in more detail, addresses the issue of heterogeneity in big data, and provides a concrete nonparametric parallelizable variable selection algorithm. Section 4 provides an in-depth analysis of the motivating Expedia dataset using the framework to conduct variable selection under different settings to determine which hotel and user characteristics influence likelihood of booking a hotel. Section 5 provides two examples on how the `MetaLP` framework provides a new understanding and resolution for problems related to Simpson's Paradox and Stein's Paradox. Section 6 provides some concluding remarks and discusses future direction of this work. Supplementary materials are also available discussing simulation study results, the relevance of MetaLP for small-data, and the `R` scripts for MapReduce implementation.

Related Literature. Several distributed learning schemes for big data have been proposed in the literature. These include a scalable bootstrap for massive data (Kleiner et. al., 2014) to assess the quality of a given estimator, a resampling-based stochastic approximation (RSA) method (Liang et. al., 2013) for analysis of a Gaussian geostatistical model, divide and recombine (D&R) approach to the analysis of large complex data (Guha et al., 2012), and also parallel algorithms for large-scale parametric linear regression proposed by Lin and Xi (2011) and Chen and Xie (2014) among others. *Instead of developing problem-specific divide-and-combine techniques, we provide a general and systematic framework that can be adapted for different varieties of "learning problems" from a nonparametric perspective, which distinguishes our work from previous proposals.* A new data analysis paradigm, combining two key ideas: LP united statistical algorithm (science of mixed data modeling), and meta-analysis (statistical basis of divide-and-combine) is proposed that can simultaneously perform nonparametric statistical *modeling and inference* under a single framework in a computationally efficient way, thus addressing a problem with much broader scope and applicability.

2. Statistical Formulation of Big Data Analysis. Our research is motivated by a real business problem of optimizing personalized web marketing for Expedia, with the goal of improving customer experience and look-to-book ratios[1] by identifying the key factors that affect consumer choices. Nearly 10 million historical hotel search and click-through transaction records selected over a window of the year 2013 are available for analysis that demonstrate the typical analytical challenges involved in big data modeling. This prototypical digital marketing case study allows us to address the following more general data modeling challenge,

[1]The look-to-book ratio is the number of people who visit a travel agency or agency web site, compared to the number who actually make a purchase. This ratio is important to online travel agents such as `Priceline.com`, `Travelocity.com`, and `Expedia.com` for determining whether their Websites are securing purchases.

which finds its applicability in many areas of modern data-driven science, engineering, and business:

How can we design nonparametric distributed algorithms that work on large amounts of data (that cannot be stored or processed by just one machine/server node) to find the most important features that affect a certain outcome of interest?

At face value, this might look as a simple two-sample inference problem that can be solved by some trivial generalization of existing 'small-data' statistical methods, but in reality, this is not the case. In fact, we are not aware of any pragmatic statistical algorithm that can achieve a similar feat. In this article we perform a thorough investigation of the theoretical and practical challenges present in the Expedia case study, and more generally in big data analysis. We emphasize the role of statistics in big data analysis and provide an overview of the three main components of our statistical theory along with the modeling challenges they are designed to overcome. In what follows, we present the conceptual building blocks of `MetaLP`–a new large-scale distributed inference tool that allows big data users to run statistical procedures on large amounts of data. Figure 1 outlines the architecture.

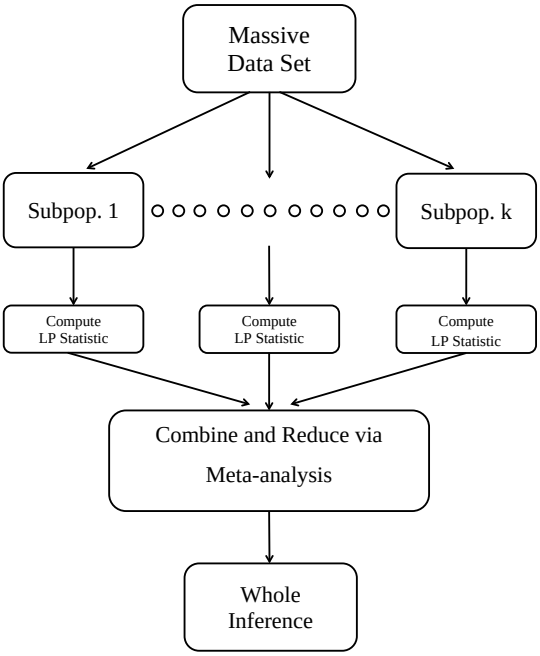

Figure 1: `MetaLP` based large-scale distributed statistical inference architecture.

2.1. *Partitioning Massive Datasets.* Dramatic increases in the size of datasets have created a major bottleneck for conducting statistical inference in a traditional "centralized" manner where we have access to the full data. The first and quite natural idea to tackle the volume problem is to divide the big data into several smaller datasets similar to the modern parallel computing database systems like Hadoop and Spark as illustrated in Figure 2. However, simply dividing the dataset does not allow data scientists to conquer the problem

of big data analysis. There are many unsettled questions that we have to carefully address using proper statistical tools to arrive at an appropriate solution.

Users must select a data partitioning scheme to split the original large data into several smaller parts and assign them to different nodes for processing. The most common technique is random partition. However, for other problems users can perform other strategies like spatial or temporal partitioning utilizing the inherent structure of the data. It may be the case that the original massive dataset is already partitioned by some natural grouping variable, in which case an algorithm that can accommodate pre-existing partitions is desirable. The number of partitions could be also defined by the user who may consider a wide range of cost metrics including the number of processors required, CPU time, job latency, memory utilization, and more when making this decision.

One important and often neglected issue associated with massive data partitioning for parallel processing is that the characteristics of the subpopulations created may vary largely. This is known as heterogeneity (Higgins and Thompson, 2002) which will be thoroughly discussed in the next section and is an unavoidable obstacle for Divide-Conquer style inference models. Heterogeneity can certainly impact data-parallel inference, but the question is can we design a method that is robust to the various data partitioning options by incorporating data-adaptive regularization? Novel exploratory diagnostics along with τ^2-regularization techniques are provided in Sections 3.5 and 3.6 to measure the severity of heterogeneity across subpopulations and adjust effect size estimates accordingly.

2.2. *LP Statistics for Mixed Data Problem.* Massive datasets typically contain a multitude of data types, and the Expedia dataset is no exception. Figure 2 shows three predictor variables in the Expedia dataset: `promotion_flag` (binary), `srch_length_of_stay` (discrete count), and `price_usd` (continuous). Even for this illustrative situation, to construct 'appropriate' statistical measures with the goal of identifying the important variables, traditional algorithmic approaches demand two pieces of information: (i) values and (ii) data-type information for every single variable X_j present in each of the subpopulations or partitioned datasets, known as the 'value-type' information-pair for statistical mining. This produces considerable complications in the computational implementation and creates serious roadblocks for building systematic and automatic algorithms for the Expedia analysis. Thus, the question of immediate concern is:

How can we develop a unified computing formula with automatic built-in adjustments that yields appropriate statistical measures without requiring the data type information from the user?

To tackle this 'variety' or mixed-data problem we design a custom-constructed discrete orthonormal polynomial-based transformation, called LP-Transformation, that provides a generic and universal representation of any random variable, defined in Section 3.1. We use this transformation technique to represent the data in a new LP Hilbert space. This data-adaptive transformation will allow us to construct unified learning algorithms by compactly expressing them as inner products in the LP Hilbert space.

2.3. *Combining Information via Confidence Distribution based Meta-analysis.* Eventually, the goal of having a distributed inference procedure critically depends on the question:

Subpopulation 1

booking_bool	promotion_flag	srch_length_of_stay	price_usd
1	0	2	164.59
0	1	7	284.48
0	1	7	123.98
⋮	⋮	⋮	⋮
0	0	4	200.46
1	0	1	194.34
0	1	3	371.27

•
•
•

Subpopulation k

booking_bool	promotion_flag	srch_length_of_stay	price_usd
1	1	1	125.65
1	0	3	149.32
0	1	8	120.63
⋮	⋮	⋮	⋮
0	1	1	224.46
1	0	1	180.64
1	1	3	174.89

Figure 2: Illustration of a partitioned data set with k subpopulations and various data types. A subset of the variables in the *Expedia* dataset are shown. The target variable Y `booking_bool`, indicates whether or not the hotel was booked. The three predictor variables shown are X_1 `promotion_flag` (indicates if sale price promotion was displayed), X_2 `srch_length_of_stay` (number of nights stayed searched), and X_3 `price_usd` (displayed price of hotel).

How to judiciously combine the "local" LP-inferences executed in parallel by different servers to get the "global" inference for the original big data?

To resolve this challenge, we make a novel connection with meta-analysis. Section 3.2 describes how we can use meta-analysis to parallelize the statistical inference process for massive datasets. Furthermore, instead of simply providing point estimates we seek to provide a distribution estimator (analogous to the Bayesian posterior distribution) for the LP-statistics via a confidence distribution (CD) that contains information for virtually all types of statistical inference (e.g. estimation, hypothesis testing, confidence intervals, etc.). Section 3.4 discusses this strategy using CD-based meta-analysis which plays a key role as a 'Combiner' to integrate the local inferences to construct a comprehensive answer for the original data. These new connections allow data scientists to fully utilize the parallel processing power of large-scale clusters for designing unified and efficient big data statistical inference algorithms.

To conclude, we have discussed the architectural overview of `MetaLP` which addresses the challenge of developing an inference framework for data-intensive applications in a way that does not require any modifications of the core statistical principles that were developed for

'small' data. Due to its simplicity and flexibility, data scientists can adapt this inference model and statistical computing philosophy to a significant number of big data problems. Next, we describe the theoretical underpinnings, algorithmic foundation, and implementation details of our data-parallel large-scale `MetaLP` inference model.

3. Elements of Distributed Statistical Learning. In this section we introduce the key foundational concepts of our proposed nonparametric distributed statistical inference paradigm. The theory connects several classical and modern statistical ideas to develop a comprehensive inference framework. We highlight along the way how these new ideas and connections address the real challenges of big data analysis as noted in Section 2.

3.1. *LP United Statistical Algorithm and Universal Representation.* An important challenge in analyzing large complex data is in the data variety problem. Developing algorithms that are generally applicable and comparable across different data types (e.g. continuous, discrete, ordinal, nominal, etc.) is an open problem that has a direct implication for developing an automatic and unified computing formula for statistical data science.

To address the data variety problem or the mixed data problem, we introduce a new nonparametric statistical data modeling framework based on an LP approach to data analysis (Mukhopadhyay and Parzen, 2014).

Data Transformation and LP Hilbert Functional Space Representation. Our approach relies on an alternative representation of the data in the LP Hilbert space, which will be defined shortly. The new representation shows how each explanatory variable, regardless of data type, can be represented as a linear combination of *data-adaptive* orthogonal LP basis functions. This data-driven transformation will allow us to construct unified learning algorithms in the LP Hilbert space. Many traditional and modern statistical measures developed for different data-types can be compactly expressed as inner products in the LP space. The following is the fundamental result for the LP basis function representation.

THEOREM 3.1 (LP representation). *Random variable X (discrete or continuous) with finite variance admits the following decomposition:* $X - \mathbb{E}(X) = \sum_{j>0} T_j(X;X)\,\mathbb{E}[XT_j(X;X)]$ *with probability* 1.

$T_j(X;X)$ for $j = 1, 2, \ldots$ are score functions constructed by Gram Schmidt orthornormalization of the powers of $T_1(X;X) = \mathcal{Z}(F^{\text{mid}}(X;X))$. Where $\mathcal{Z}(X) = (X - \mathbb{E}[X])/\sigma(X)$, $\sigma^2(X) = \text{Var}(X)$, and the mid-distribution transformation of a random variable X is defined as

$$F^{\text{mid}}(x;X) = F(x;X) - .5p(x;X),\ p(x;X) = \Pr[X = x],\ F(x;X) = \Pr[X \le x]. \tag{3.1}$$

We construct the LP score functions on $0 < u < 1$ by letting $x = Q(u;X)$, where $Q(u;X)$ is the quantile function of the random variable X

$$S_j(u;X) \;=\; T_j(Q(u;X);X),\quad Q(u;X) = \inf\{x : F(x) \ge u\}. \tag{3.2}$$

Why is it called the LP-basis? Note that our specially designed basis functions vary naturally according to data type unlike the fixed Fourier and wavelet bases as shown in Figure

5. Note an interesting similarity of the shapes of LP score functions and shifted Legendre Polynomials for the *continuous* feature `price_usd`. In fact, as the number of atoms (# distinct values) of a random variable $A(X) \to \infty$ (moving from discrete to continuous data type) the shape converges to the smooth Legendre Polynomials. To emphasize this universal limiting shape we call it **L**egendre-**P**olynomial-like (**LP**) orthogonal basis. For any general X, LP-polynomials are piecewise-constant orthonormal functions over $[0, 1]$ as shown in Figure 5. This data-driven property makes the LP transformation uniquely advantageous in constructing a generic algorithmic framework to tackle the mixed-data problem.

Constructing Measures by LP Inner Product. Define the two-sample LP statistic for variable selection of a *mixed random* variable X (either continuous or discrete) based on our specially designed score functions

$$\mathrm{LP}[j; X, Y] \;=\; \mathrm{Cor}[T_j(X; X), Y] \;=\; \mathbb{E}[T_j(X; X)T_1(Y; Y)]. \tag{3.3}$$

To prove the second equality of Equation (3.3) (which expresses our variable selection statistic as LP-inner product measure) verify the following for Y binary

$$\mathcal{Z}(y; Y) \;=\; T_1(y; Y) \;=\; \begin{cases} -\sqrt{\dfrac{p}{1-p}} & \text{for } y = 0 \\ \sqrt{\dfrac{1-p}{p}} & \text{for } y = 1. \end{cases}$$

LP statistic properties. Using empirical process theory we can show that the sample LP-Fourier measures $\sqrt{n}\widetilde{\mathrm{LP}}[j; X, Y]$ asymptotically converge to i.i.d standard normal distributions (Mukhopahyay and Parzen, 2014).

As an example of the power of LP-unification, we describe $\mathrm{LP}[1; X, Y]$ that systematically reproduces all the traditional linear statistical variable selection measures for different data types of X. Note that the Nonparametric Wilcoxon method to test the equality of two distributions can equivalently be represented as $\mathrm{Cor}(\mathbb{I}\{Y = 1\}, F^{\mathrm{mid}}(X; X))$ which leads to the following important alternative LP representation result.

THEOREM 3.2. *Two sample Wilcoxon Statistic W can be computed as*

$$W(X, Y) \;=\; \mathrm{LP}[1; X, Y]. \tag{3.4}$$

Our computing formula for the Wilcoxon statistic using $\mathrm{LP}[1; X, Y]$ offers automatic *built-in adjustments for data with ties*; hence no further tuning is required.

Furthermore, if we have X and Y both binary, (i.e. they form a 2×2 table), then we have

$$\begin{aligned} T_1(0; X) &= -\sqrt{P_{2+}/P_{1+}}, \quad T_1(1; X) = \sqrt{P_{1+}/P_{2+}} \\ T_1(0; Y) &= -\sqrt{P_{+2}/P_{+1}}, \quad T_1(1; Y) = \sqrt{P_{+1}/P_{+2}}, \end{aligned} \tag{3.5}$$

where $P_{i+} = \sum_j P_{ij}$ and $P_{+j} = \sum_i P_{ij}$, and P_{ij} denotes the 2×2 probability table and

$$\begin{aligned} \mathrm{LP}[1; X, Y] \;&=\; \mathbb{E}[T_1(X; X)T_1(Y; Y)] \;=\; \sum_{i=1}^{2}\sum_{j=1}^{2} P_{ij}T_1(i-1; X)\,T_1(j-1; Y) \\ &=\; \big(P_{11}P_{22} - P_{12}P_{21}\big)/\big(P_{1+}P_{+1}P_{2+}P_{+2}\big)^{1/2}. \end{aligned} \tag{3.6}$$

Following result summarizes the observation in (3.6).

THEOREM 3.3. *For 2×2 contingency table with Pearson correlation ϕ we have,*

$$\phi(X, Y) \;=\; \mathrm{LP}[1; X, Y]. \tag{3.7}$$

Beyond Linearity. High order Wilcoxon statistics are LP-statistics of high order score functions $T_j(X; X)$, which detect the *distributional* differences as in variability, skewness, or tail behavior for two different classes. The LP-statistics $\mathrm{LP}[j; X, Y], j > 1$ capture how the distribution of a variable changes over classes in a systematic way, applicable for mixed-data types. We are *not* aware of any other statistical technique that can achieve similar goals.

To summarize, the remarkable property of LP-statistics is that it allows data scientists to write a *single* computing formula for any variable X, irrespective of its data type, with a *common* metric and asymptotic characteristics. This leads to a huge practical benefit in designing a unified method for combining 'distributed local inferences' without requiring the data-type information for the variables in our partitioned dataset.

3.2. *Meta-Analysis and Data-Parallelism.* The objective of this section is to provide a new way of thinking about the problem: how to appropriately combine "local" inferences to derive reliable and robust conclusions for the original large dataset? This turns out to be one of the most crucial and heavily *neglected* part of data-intensive modeling that decides the fate of big data inference. Here we introduce the required statistical framework that can answer the key question: *how to compute individual weights for each partitioned dataset?* Our framework adopts the concept of meta-analysis to provide a general recipe for constructing such algorithms for large-scale parallel computing. This will allow us to develop statistical algorithms that can judiciously balance computational speed and statistical accuracy in data analysis.

Brief Background on Meta-Analysis. Meta-analysis (Hedges and Olkin, 1985) is a statistical technique by which information from independent studies is assimilated, which has its origins in clinical settings. It was invented primarily to combat the problem of under-powered 'small data' studies. A key benefit of this approach is the aggregation of information leading to a higher statistical power as opposed to less precise inference derived from a single small data study. A huge amount of literature exists on meta-analysis, including a careful review of recent developments written by Sutton and Higgins (2008) which includes 281 references.

Relevance of Meta-analysis for big data inference? First note that unlike the classical situation, we don't have any low statistical power issue for big data problems. At the same time we are unable to analyze the whole dataset all-at-once using a single machine in a classical inferential setup. Our novelty lies in *recognizing* that meta-analytic logic provides a formal statistical framework to rigorously formulate the problem of combining distributed inferences. We apply meta-analysis from a completely different perspective and motivation, as a tool to facilitate distributed inference for very large datasets. This novel connection provides a statistically sound powerful mechanism to combine 'local' inferences by properly determining the 'optimal' weighting strategy (Hedges and Olkin, 1985).

We partition big data systematically into several subpopulations (small datasets) over a distributed database, estimate parameters of interest in each subpopulation separately, and

then combine results using meta-analysis as demonstrated in Figure 1 . Thus meta-analysis provides a methodical way to pool the information from all of the small subpopulations and produce a singular powerful combined inference for the original large dataset. In some circumstances, the dataset may already be partitioned (each group could be an image or a large text document) and stored in different servers based on some reasonable partitioning scheme. Our distributed statistical framework can work with these predefined groupings as well by combining them using the meta-analysis framework to arrive at the final combined inference.

We call this statistical framework, which utilizes both LP statistics and meta-analysis methodology, as `MetaLP`, and it consists of two parts: (i) the LP statistical map function or algorithm (that tackles the 'variety' problem), and (ii) the meta-analysis methodology for merging the information from all subpopulations to get the final inference.

3.3. *Confidence Distribution and LP Statistic Representation.* The Confidence Distribution (CD) is a distribution estimator, rather than a point or interval estimator, for a particular parameter of interest. From the CD, all the traditional forms of statistical estimation and inference (e.g. point estimation, confidence intervals, hypothesis testing) can be produced. Hence CDs contain a wealth of information about parameters of interest which makes them attractive for understanding key parameters. Moreover, CDs can be utilized within the meta-analysis framework, as we will show in the next section, which enables our algorithm to provide a variety of classical statistics mentioned previously for users. More specifically, the CD is a concept referring to a sample-dependent distribution function on the parameter space that can represent confidence intervals of all levels for a parameter of interest. Xie and Singh (2013) gave a comprehensive review of the CD concept and emphasized that the CD is a frequentist statistical notion. Schweder and Hjort (2002) first defined the CD formally and Singh, Xie, and Strawderman (2005) extended the CD concept to asymptotic confidence distributions (aCDs):

Definition 3.1. Suppose Θ is the parameter space of the unknown parameter of interest θ, and ω is the sample space corresponding to data $\mathbf{X}_n = \{X_1, X_2, \ldots, X_n\}^T$. Then a function $H_n(\cdot) = H_n(\mathbf{X}, \cdot)$ on $\omega \times \Theta \to [0, 1]$ is a confidence distribution (CD) if: (i). For each given $\mathbf{X}_n \in \omega$, $H_n(\cdot)$ is a continuous cumulative distribution function on Θ; (ii). At the true parameter value $\theta = \theta_0$, $H_n(\theta_0) = H_n(\mathbf{X}, \theta_0)$, as a function of the sample $\mathbf{X}_n$, following the uniform distribution $U[0, 1]$. The function $H_n(\cdot)$ is an asymptotic CD (aCD) if the $U[0, 1]$ requirement holds only asymptotically for $n \to \infty$ and the continuity requirement on $H_n(\cdot)$ can be relaxed.

By definition, the CD is a function of both a random sample and the parameter of interest. (i) The definition requires that for each sample, the CD should be a distribution function on the parameter space. The $U[0, 1]$ requirement in (ii) allows us to construct confidence intervals from a CD easily, meaning that $(H_n^{-1}(\alpha_1), H_n^{-1}(1 - \alpha_2))$ is a $100(1 - \alpha_1 - \alpha_2)\%$ confidence interval for the parameter θ_0 for any $\alpha_1 > 0$, $\alpha_2 > 0$, and $\alpha_1 + \alpha_2 < 1$.

Generally, the CD can be easily derived from the *stochastic internal representation* (Parzen, 2013) of a random variable and a pivot $\Psi(S\ ,\ \theta)$ whose distribution does not depend on the parameter θ, where θ is the parameter of interest and S is a statistic derived from the data. Here, we derive the CD for the LP statistic. Suppose $\widehat{\mathrm{LP}}[j; X, Y]$ is the estimated jth LP

statistic for the predictor variable X and binary response Y. The limiting asymptotic normality of the empirical LP-statistic can be compactly represent as:

$$\text{LP}[j;X,Y] \,\Big|\, \widehat{\text{LP}}[j;X,Y] \quad = \quad \widehat{\text{LP}}[j;X,Y] + \frac{Z}{\sqrt{n}}, \tag{3.8}$$

which is the stochastic internal representation (similar to the stochastic differential equations representation) of the LP statistic. Thus we have the following form of the confidence distribution, which is the cumulative distribution function of $\mathcal{N}\big(\widehat{\text{LP}}[j;X,Y], 1/n\big)$:

$$H_\Phi(\text{LP}[j;X,Y]) = \Phi\left(\sqrt{n}\left(\text{LP}[j;X,Y] - \widehat{\text{LP}}[j;X,Y]\right)\right), \tag{3.9}$$

The above representation satisfies the conditions in the CD definition and therefore is the CD of $\text{LP}[j;X,Y]$. Since, our derivation is based on assumption that $n \to \infty$, the CD we just derived is the asymptotic CD.

3.4. *Confidence Distribution-based Meta-Analysis.* Using the theory presented in Section 3.3 we can estimate the confidence distribution (CD) for the LP statistics for each of the k subpopulations, $H(\text{LP}_\ell[j;X,Y])$, and the corresponding point estimators $\widehat{\text{LP}}_\ell[j;X,Y]$ for $\ell = 1,\ldots,k$. The next step of our `MetaLP` algorithm is to judiciously combine information contained in the CDs for all subpopulations to arrive at the combined CD $H^{(c)}(\text{LP}[j;X,Y])$ based on the whole dataset for that specific variable X. The framework relies on a confidence distribution-based unified approach to meta-analysis introduced by Singh, Xie, and Strawderman (2005). The combining function for CDs across k different studies can be expressed as:

$$H^{(c)}(\text{LP}[j;X,Y]) \;=\; G_c\{g_c(H(\text{LP}_1[j;X,Y]),\ldots,H(\text{LP}_k[j;X,Y])\}. \tag{3.10}$$

The function G_c is determined by the monotonic g_c function defined as

$$G_c(t) = P(g_c(U_1,\ldots,U_k) \le t),$$

in which $U_1,\ldots,U_k$ are independent $U[0,1]$ random variables. A popular and useful choice for g_c is

$$g_c(u_1,\ldots,u_k) = \alpha_1 F_0^{-1}(u_1) + \ldots + \alpha_k F_0^{-1}(u_k), \tag{3.11}$$

where $F_0(\cdot)$ is a given cumulative distribution function and $\alpha_\ell \ge 0$, with at least one $\alpha_\ell \ne 0$, are generic weights. $F_0(\cdot)$ could be any distribution function, which highlights the flexibility of the proposed framework. Hence the following theorem introduces a reasonable proposed form of the combined aCD for $\text{LP}[j;X,Y]$.

THEOREM 3.4. *Setting $F_0^{-1}(t) = \Phi^{-1}(t)$ and $\alpha_l = \sqrt{n_\ell}$, where n_ℓ is the size of subpopulation $\ell = 1,\ldots,k$, the following combined aCD for $\text{LP}[j;X,Y])$ follows:*

$$H^{(c)}(\text{LP}[j;X,Y]) = \Phi\left[\left(\sum_{\ell=1}^{k} n_\ell\right)^{1/2}\left(\text{LP}[j;X,Y] - \widehat{\text{LP}}^{(c)}[j;X,Y]\right)\right] \quad \text{with} \tag{3.12}$$

$$\widehat{\mathrm{LP}}^{(c)}[j;X,Y] = \frac{\sum_{\ell=1}^{k} n_\ell \widehat{\mathrm{LP}}_\ell[j;X,Y]}{\sum_{\ell=1}^{k} n_\ell} \tag{3.13}$$

where $\widehat{\mathrm{LP}}^{(c)}[j;X,Y]$ *and* $\left(\sum_{\ell=1}^{k} n_\ell\right)^{-1}$ *are the mean and variance respectively of the combined aCD for* $\mathrm{LP}[j;X,Y]$.

To prove this theorem verify that replacing $H(\mathrm{LP}_\ell(j;X,Y))$ by (3.9) in Equation (3.10) along with the choice of combining function given in (3.11), where $F_0^{-1}(t) = \Phi^{-1}(t)$ and $\alpha_\ell = \sqrt{n_\ell}$ we have

$$H^{(c)}(\mathrm{LP}[j;X,Y]) = \Phi\left[\frac{1}{\sqrt{\sum_{\ell=1}^{k} n_\ell}} \sum_{\ell=1}^{k} \sqrt{n_\ell} \frac{\mathrm{LP}[j;X,Y] - \widehat{\mathrm{LP}}_\ell[j;X,Y]}{1/\sqrt{n_\ell}}\right].$$

3.5. *Diagnostic of Heterogeneity.* Our approach allows parallel data processing by dividing the original large dataset into several subpopulations and then finally combining all the information under a confidence distribution based LP meta-analysis framework. One danger that can potentially arise from this Divide-Combine-Conquer style big data analysis is the heterogeneity problem. The root cause of this problem is different characteristics across the subpopulations. It is arguably one of the most significant roadblocks, which is often ignored, for analyzing distributed data. Failure to take heterogeneity into account can easily spoil the big data discovery process.

This heterogeneity across subpopulations will produce very different statistical estimates which may not faithfully reflect the original parent dataset. So it is of the utmost importance that we should diagnose and quantify the degree to which each variable suffers from heterogeneous subpopulation groupings. We resolve this challenge by using the I^2 statistic (Higgins and Thompson, 2002), which is introduced next.

Define Cochran's Q statistic:

$$Q = \sum_{\ell=1}^{k} \alpha_\ell \left(\widehat{\mathrm{LP}}_\ell[j;X,Y] - \widehat{\mathrm{LP}}^{(c)}[j;X,Y]\right)^2, \tag{3.14}$$

where $\widehat{\mathrm{LP}}_\ell[j;X,Y]$ is the estimated LP-statistic from subpopulation ℓ, α_ℓ is the weight for subpopulation ℓ as defined in Equation 3.10, and $\widehat{\mathrm{LP}}^{(c)}[j;X,Y]$ is the combined meta-analysis estimator. Compute the I^2 statistic by

$$I^2 = \begin{cases} \frac{Q-(k-1)}{Q} \times 100\% & \text{if } Q > (k-1); \\ 0 & \text{if } Q \le (k-1); \end{cases} \tag{3.15}$$

where k is the number of subpopulations. A rule of thumb in practice is that, $0\% \le I^2 \le 40\%$ indicates heterogeneity among subpopulations is not severe.

3.6. τ^2 *Regularization to Tackle Heterogeneity in Big Data.* The variations among the subpopulations impact LP statistic estimates, which are not properly accounted for in the Theorem 3.4 model specification. This is especially severe for big data analysis as it is very likely that a substantial number of variables may be affected by heterogeneity across subpopulations (however the curse of heterogeneity can also be present in small data as shown in Supplementary Section B. To better account for the heterogeneity in the distributed statistical inference framework, following Hedges and Olkin (1985, p. 123), we introduce an additional parameter (τ^2) to account for uncertainty due to heterogeneity across subpopulations. A hierarchical structure is added in this model:

$$\widehat{\mathrm{LP}}_\ell[j;X,Y] \;\Big|\; \mathrm{LP}_\ell[j;X,Y], s_i \overset{\text{iid}}{\sim} N(\mathrm{LP}_\ell[j;X,Y], s_i^2), \text{ and} \tag{3.16}$$

$$\mathrm{LP}_\ell[j;X,Y] \;\Big|\; \mathrm{LP}[j;X,Y], \tau \overset{\text{iid}}{\sim} N(\mathrm{LP}[j;X,Y], \tau^2), \quad \ell = 1,\ldots,k. \tag{3.17}$$

Under the new model specification, the CD of the LP statistic for the ℓ-th group is $H(\mathrm{LP}_\ell[j;X,Y]) = \Phi((\mathrm{LP}[j;X,Y] - \widehat{\mathrm{LP}}_\ell[j;X,Y])/(\tau^2 + s_\ell^2)^{1/2})$ where $s_\ell = 1/\sqrt{n_\ell}$. The following theorem provides the form of the combined aCD under this specification.

THEOREM 3.5. *Setting* $F_0^{-1}(t) = \Phi^{-1}(t)$ *and* $\alpha_\ell = 1/\sqrt{(\tau^2 + (1/n_\ell))}$, *where* n_ℓ *is the size of subpopulation* $\ell = 1,\ldots,k$, *the following combined aCD for* $\mathrm{LP}[j;X,Y]$ *follows:*

$$H^{(c)}(\mathrm{LP}[j;X,Y]) = \Phi\left[\left(\sum_{\ell=1}^{k} \frac{1}{\tau^2 + (1/n_\ell)}\right)^{1/2} (\mathrm{LP}[j;X,Y] - \widehat{\mathrm{LP}}^{(c)}[j;X,Y])\right] \quad \text{with} \tag{3.18}$$

$$\widehat{\mathrm{LP}}^{(c)}[j;X,Y]) = \frac{\sum_{\ell=1}^{k} (\tau^2 + (1/n_\ell))^{-1} \widehat{\mathrm{LP}}_\ell[j;X,Y])}{\sum_{\ell=1}^{k} (\tau^2 + (1/n_\ell))^{-1}} \tag{3.19}$$

where $\widehat{\mathrm{LP}}^{(c)}[j;X,Y])$ *and* $(\sum_{\ell=1}^{k} 1/(\tau^2 + (1/n_\ell)))^{-1}$ *are the mean and variance respectively of the combined aCD for* $\mathrm{LP}[j;X,Y]$.

The proof is similar to that for Theorem 3.1. The DerSimonian and Laird (DerSimonian and Laird, 1986) and restricted maximum likelihood estimators of the data-adaptive heterogeneity regularization parameter τ^2 are given in Supplementary Section D.

4. Expedia Personalized Hotel Search Dataset. `MetaLP` is a generic distributed inference platform that can scale to very large datsets. Motivated by `MetaLP` here we develop a *model-free parallelizable two-sample algorithm* (assuming no model of any form and requiring no nonparametric smoothing) under the big data inference paradigm and apply it for the Expedia digital marketing problem. Detailed discussion on each one of the following components of our big data two-sample inference model is given in the next sections:

- (Section 4.1) Data Description.
- (Section 4.2) Data Partitioning.

- (Section 4.3) LP Map Function.
- (Section 4.4) Heterogeneity Diagnostic and Regularization.
- (Section 4.5) Meta Reducer via LP-Confidence Distribution.
- (Section 4.6) Robustness Study.

4.1. *Data Description.* Expedia provided this dataset of 10 million hotel search results, collected over a window of the year 2013; online customers input a list of search criteria, such as length of stay, booking window, and number of children, to the Expedia website as shown in Figure 3(a). Based on the search criteria and customer information, Expedia sends back a ranked list of available hotels that customers can book for their travels (see Figure 3(b)). Given the list, customers behaviors (click, book, or ignore) were then recorded by Expedia. The question of significant business value to the Expedia digital marketing team: which factors of these "search result impressions" (search criteria, hotel characteristics, customer information, competitor information) are most closely related to booking behavior, which could be used to increase user engagement, travel experience, and search personalization.

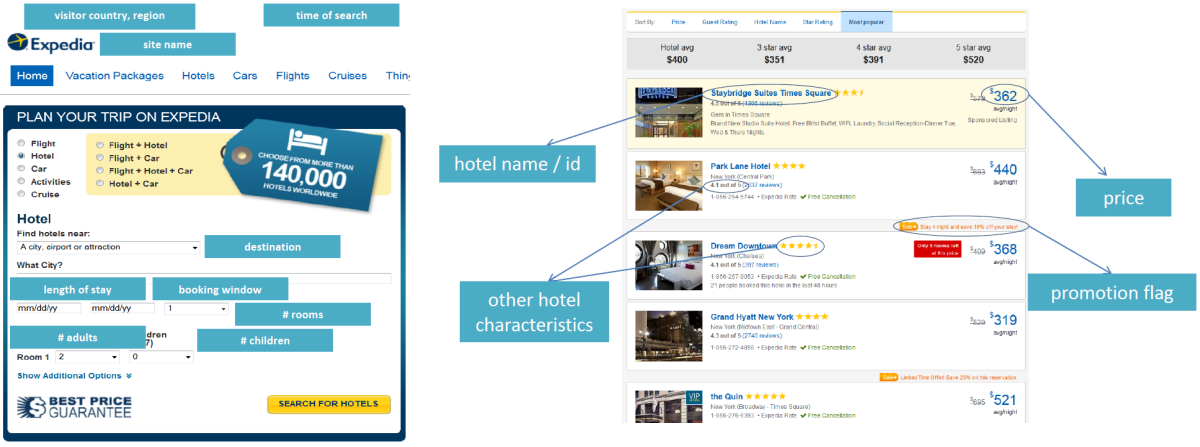

Figure 3: (color online) Left: (a) Search window with search criteria variables; Right: (b) List of ranked hotels returned by Expedia with hotel characteristic variables.

The dataset contains a huge number of observations (399,344 unique search lists and 9,917,530 observations) each with 45 predictor variables of various of data types (e.g. categorical, binary, and continuous). There are various variables in the dataset related to user characteristics (e.g. visitor location, search history, etc.), search criteria (e.g. length of stay, number of children, room count, etc.), static hotel information (e.g. star rating, hotel location, historic price, review scores, etc.), dynamic hotel information (e.g. current price, promotion flag, etc), and competitor's information (e.g. price difference and availability), that may impact users' booking behaviors. The response variable, `booking_bool`, is a binary variable that indicates whether the hotel was booked or not. The remaining variables contain the variables mentioned previously. Descriptions of some representative variables and their data types are presented in Table 1. A complete list of the variables can be found on `Kaggle`'s website.

Variable Category	Variables	Data Type	Description
User's Information	visitor_location_country_id	Discrete	The ID of the country in which the customer is located
	visitor_hist_starrating	Continuous	The mean star rating of hotels the customer has previously purchased
	visitor_hist_adr_usd	Continuous	The mean price of the hotels the customer has previously purchased
	orig_destination_distance	Continuous	Physical distance between the hotel and the customer
Search Criteria	srch_length_of_stay	Discrete	Number of nights stay that was searched
	srch_booking_window	Discrete	Number of days in the future the hotel stay started
	srch_adults_count	Discrete	The number of adults specified in the hotel room
	srch_children_count	Discrete	The number of children specified in the hotel room
	srch_room_count	Discrete	Number of hotel rooms specified in the search
	srch_saturday_night_bool	Binary	If short stay including Saturday night
Static hotel characteristics	prop_country_id	Discrete	The ID of the country the customer is located
	prop_starrating	Discrete	The star rating of the hotel
	prop_review_score	Continuous	The mean customer review score for the hotel
	prop_location_score1	Continuous	Desirability of hotel location (1)
	prop_location_score2	Continuous	Desirability of hotel location (2)
	prop_log_historical_price	Continuous	Mean price of the hotel over the last trading period
	pprop_brand_bool	Discrete	If independent or belongs to a hotel chain
Dynamic hotel characteristics	price_usd	Continuous	Displayed price of the hotel for the given search
	promotion_flag	Discrete	If the hotel had a sale price promotion
	gross_booking_usd	Continuous	Total value of the transaction
Competitor's Information	comp1_rate_percent_dif	Continuous	The absolute percentage difference between competitors
	comp1_inv	Binary	If competitor 1 has hotel availability
	comp1_rate	Discrete	If Expedia has lower/same/higher price than competitor X
Other Information	srch_id	Discrete	The ID of the search
	site_id	Discrete	ID of the Expedia point of sale
Response Variables	booking_bool	Binary	If the hotel is booked

TABLE 1

Data description for Expedia dataset. The column 'data type' indicates the presence of variety problem

4.2. *Partition.* We consider two different partitioning schemes that are reasonable for the Expedia dataset: random partitioning, in which we get subpopulations with similar sizes that are relatively homogeneous, and predefined partitioning, in which the subpopulations are relatively heterogeneous with contrasting sizes.

Step 1. We randomly assign search lists, which are collections of observations from same search id in the dataset, to 200 different subpopulations. Random assignment of search lists rather than individual observations ensures that sets of hotels viewed in the same search session are all contained in the same subpopulation. The number of subpopulations chosen here can be adapted to meet the processing and time requirements of different users (e.g. users with more servers available may choose to increase the number of subpopulations to take advantage of the additional processing capability).

For example, we can randomly partition the dataset into $S = 50, 100, 150, 200, ...$ subpopulations. We show in Section 4.6 that our method is robust to different numbers of subpopulations. There may be situations where we already have 'natural' groupings in the dataset, which can be directly utilized as subpopulations. For example, consider the scenario where the available Expedia data are collected from different countries by variable `visitor_location_country_id`, a indicator of visitor's location (country). In this setting, our framework can directly utilize these predetermined subpopulations for processing rather than having to pull it all together and randomly assign subpopulations. Often, this partition scheme may result in heterogeneous subpopulations. For example, in the Expedia dataset almost half of the observations are from country 207 (possibly the U.S.). Thus, extra steps must be taken to deal with this situation as described in Section 4.4.

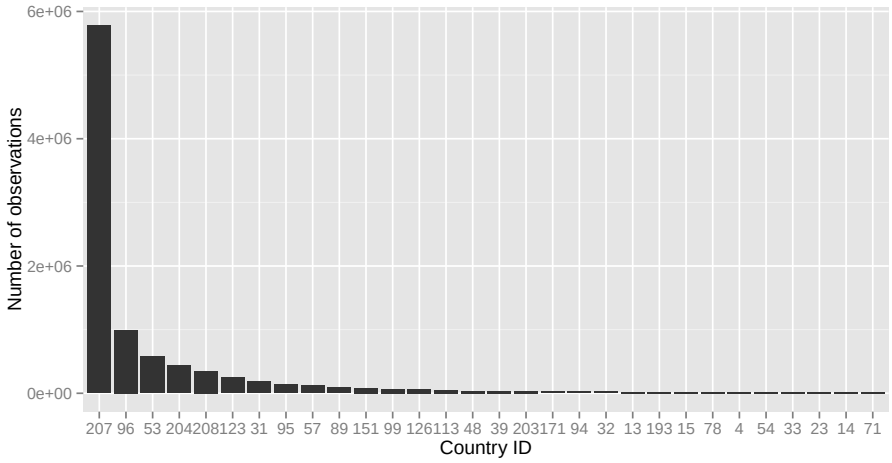

Figure 4: Barplot: Number of observations for top 40 largest subpopulations from partitioning by `visitor_location_country_id`

Figure 4 shows the number of observations for the top 40 largest subpopulations from partitioning by `visitor_location_country_id`. The top 3 largest groups contain 74% of the total observations. Group 207 contains almost 50% of the total observations. On the other hand, random partitioning results in roughly equal sample size across subpopulations (about 49,587 observations each).

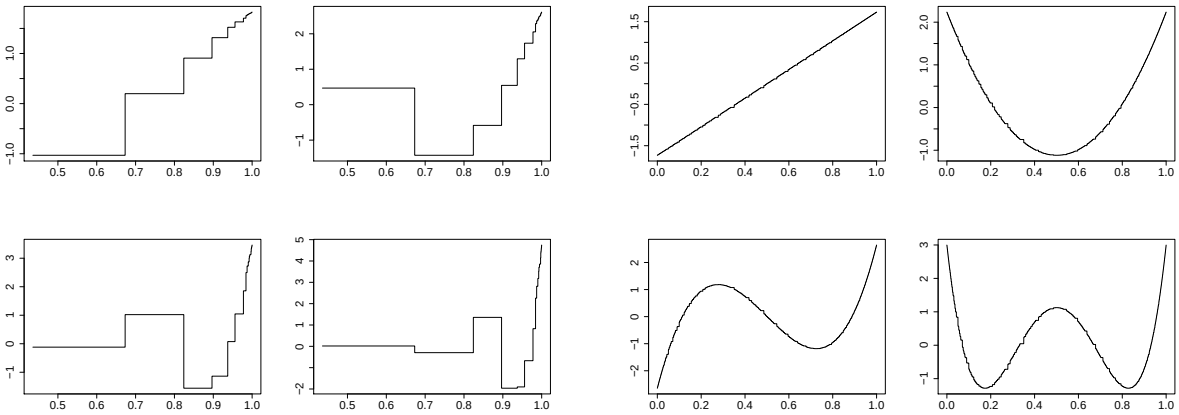

Figure 5: (a) Left panel shows the shape of the first four LP orthonormal score functions for the variable `variable_length_of_stay`, which is a discrete random variable taking values $0, \ldots, 8$; (b) Right: the shape of the LP basis for the continuous `price_usd`. As the number of atoms (# distinct values) of a random variable increases (moving from discrete to continuous data type) the shape of our custom designed score polynomials automatically approaches to (by construction) a universal shape, close to shifted **L**egendre-**P**olynomials over $(0, 1)$.

4.3. *LP Map Function.* We tackle the existing variety problem (see Table 1) by developing automated mixed-data algorithms using LP-statistical data modeling tools.

Step 2. Following the theory in Section 3.1, construct LP-score polynomials $T_j(x; X_i)$ for each variable based on each partitioned input dataset. Figure 5 shows the shapes of LP-basis polynomials for variables `variable_length_of_stay` (discrete variable) and `price_usd` (continuous variable).

Step 3. Estimate the $\mathrm{LP}_\ell[j; X_i, Y]$ statistics (which denotes the jth LP statistics for the ith variable in the ℓth subpopulation)

$$\widetilde{\mathrm{LP}}_\ell[j; X_i, Y] \;=\; n_\ell^{-1} \sum_{k=1}^{n_\ell} T_j(x_k; X_i)\, T_1(y_k; Y). \tag{4.1}$$

Step 4. Compute the corresponding LP-confidence distribution given by

$$\Phi\left(\sqrt{n}\left(\mathrm{LP}_\ell[j; X_i, Y] - \widehat{\mathrm{LP}}_\ell[j; X_i, Y]\right)\right),$$

for each of the 45 variables across the 200 random subpopulations (or 233 predefined subpopulations defined by the grouping variable `visitor_location_country_id`), where $i = 1, \ldots, 45$, $\ell = 1, \ldots, 200$, and i and ℓ are the indexes for variable and subpopulation respectively. The estimator values $\widehat{\mathrm{LP}}_\ell[j; X_i, Y]$ and n_ℓ (used to find standard deviation) are stored for use in the next step.

4.4. *Heterogeneity: Diagnostic and Regularization.* Figure 6 shows the first order LP-statistic of the variable `price_usd` across different subpopulations based on random and

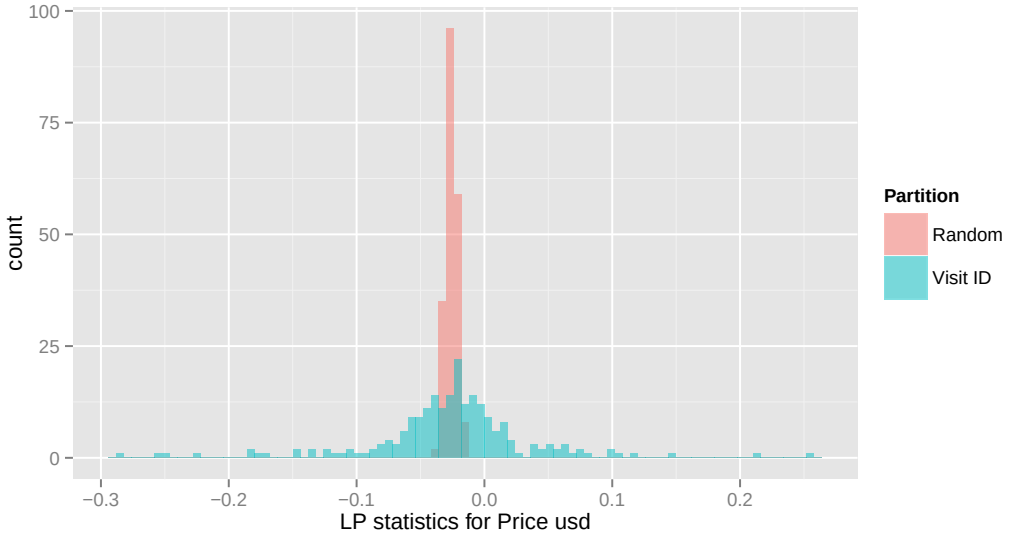

Figure 6: (color online) Histogram of LP-statistic of the variable `price_usd` based on random partition and `visitor_location_country_id` partition.

`visitor_location _country_id` partition schemes. It is clear that the random partition produces relatively homogeneous LP-estimates as the distribution is much more concentrated or clustered together. On the other hand, `visitor_loc ation_country_id` partition results in more heterogeneous LP statistics, which is reflected in the histogram. In fact The standard deviation of LP statistics for `visitor_location_country_id` partition is about 15 times more than that of the random partition, which further highlights the underlying heterogeneity issue. Thus care must be taken to incorporate this heterogeneity in a judicious manner that ensures consistent inference. We advocate the method mentioned in Section 3.5.

Step 5. Compute the Cochran's Q-statistic using (3.14) and I^2 heterogeneity index (3.15) based on $\mathrm{LP}_1[j; X_i, Y], \ldots, \mathrm{LP}_S[j; X_i, Y]$ for each i and j. For the random partitioning scheme, our subpopulations are fairly homogeneous (with respect to all variables) as all I^2 statistics are below 40% (see Figure 7(a)); on the other hand, `visitor_location_country_id` based predefined partitioning divides data into heterogeneous subpopulations for some variables as shown in Figure 7(b) (i.e. some variables have I^2 values (red dots) outside of the permissible range of 0 to 40%).

Step 6. Compute the DerSimonian and Laird data-driven estimate

$$\hat{\tau}_i^2 = \max\left\{0, \frac{Q_i - (k-1)}{n - \sum_\ell n_\ell^2/n}\right\}, \qquad (i = 1, \ldots, p).$$

One can also use other enhanced estimators like the restricted maximum-likelihood (REML) estimator as discussed in Supplementary Section D. The I^2 diagnostic *after* τ^2 regularization is shown in Figure 7(b) (blue dots), which suggest that all I^2 values after regularization fall within the acceptable range of 0 to 40%. The results suggests that our framework are able to deal with heterogeneity issues among subpopulations, as we can perform τ^2 regularization

when subpopulations appear to be heterogeneous, which protects the validity of the meta-analysis approach.

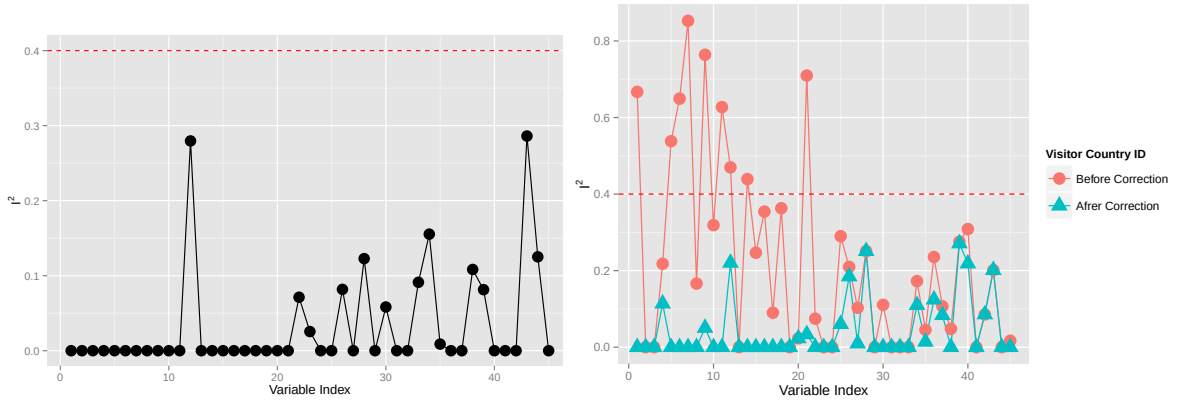

Figure 7: (color online) (a) I^2 Diagnostic for randomly partitioned subpopulations; (b) Pre-determined grouping: comparison of I^2 diagnostics between before τ correction (red dots) and after τ correction (blue dots).

4.5. *Meta Reducer Step.* This step combines estimates and confidence distributions of LP statistics from different subpopulations to estimate the combined confidence distribution of the LP statistic for each variable as outlined in Section 3.6.

Step 7. Use τ-corrected weights to properly taking into account the heterogeneity effect. Compute $\widehat{\mathrm{LP}}^{(c)}[j; X, Y])$ by (3.19) and the corresponding LP-confidence distribution using Theorem 3.5.

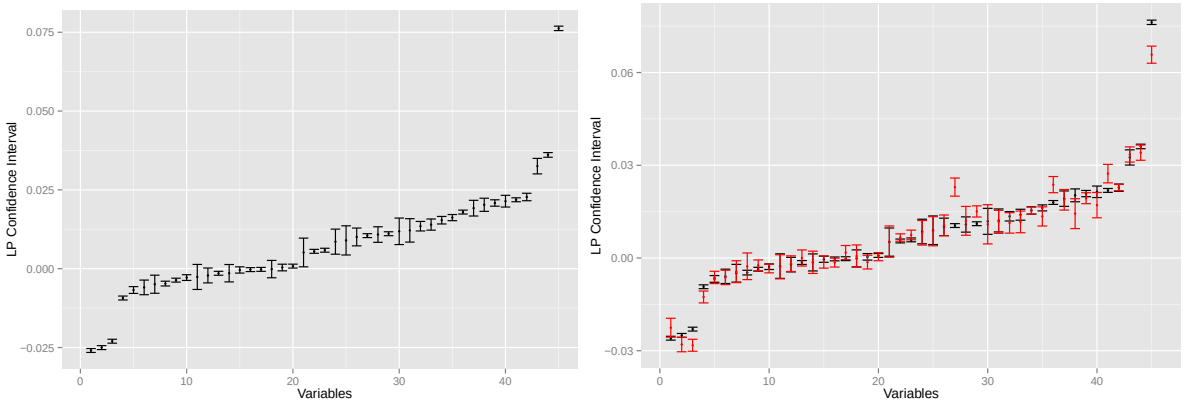

Figure 8: (color online) (a) Expedia Data: 95 % Confidence Intervals for each variables' LP Statistics; (b) 95% Confidence Interval for Random Sampling Partitioning (black) and country ID Partitioning (red).

The results for random subpopulation assignment can be found in Figure 8(a). Variables with indexes 43, 44, and 45 have highly significant positive relationships with `booking_bool`, the binary response variable. Those variables are `prop_location_score2`, the second score

Rank	Random partition	Predetermined partition
1	prop_location_score2	prop_location_score2
2	promotion_flag	promotion_flag
3	price_usd	price_usd
4	srch_length_of_stay	srch_length_of_stay
5	prop_starring	srch_query_affinity_score

TABLE 2
Top five influential variables by random partition and predetermined partition

outlining the desirability of a hotel's location, promotion_flag, if the hotel had a sale price promotion specifically displayed, and srch_query_affinity_score, the log of the probability a hotel will be clicked on in Internet searches; there are three variables that have highly negative impacts on hotel booking: price_usd, displayed price of the hotel for the given search, srch_length_of_stay, number of nights stay that was searched, and srch_booking_window, number of days in the future the hotel stay started from the search date. Moreover, there are several variables' LP statistics whose confidence intervals include zero, which means those variables have an insignificant influence on hotel booking. The top five most influential variables in terms of absolute value of LP statistic estimates are prop_location_score2, promotion flag, price_usd, srch_length_of_stay, and prop_starring.

If we apply the same algorithm to the predefined partitioned dataset and compare the top five influential variables with those from the randomly partitioned dataset, 4 out of 5 are the same, while prop_starring changed to srch_query_affinity_score. A comprehensive comparison between the two lists are shown in Figure 8(b). It provides a comparison of 95% confidence intervals for LP statistics across each variable based on randomly assigned subpopulations and predefined subpopulations by visitor country. Here, it is shown how the heterogeneity from the predetermined subpopulations impacts the statistical inference slightly in terms of wider confidence intervals. However, across the majority of the variables, the inference obtained from randomly assigned subpopulations and the predetermined subpopulations are largely consistent.

The top five most influential variables from both partition schemes are reported in Table 2. All variables selected by our algorithm are intuitive. For instance, prop_location_score indicates the desirability of the hotel location. if the hotel's location has higher desirability score, users tend to book the hotel more frequently; promotion_flag indicates if the hotel has special promotion. The probability the hotel is booked will be higher if the hotel has a special discount or promotion.

4.6. *Robustness to Size and Number of Subpopulations.* Due to different capabilities of computing systems available to users, users may choose different sizes and numbers of subpopulations for distributed computing. This requires our algorithm to be robust to numbers and sizes of subpopulations. To assess this robustness, we develop multiple random partitions of the whole dataset into $S = 50, 100, 150, 200, 250, 300, 350, 400, 450, 500$ subpopulations respectively, and then show that we are able to get consistent combined LP estimators even with big differences in numbers and sizes of subpopulations. We examine prop_location_score2, promotion_flag, and price_usd as the three most influential variables; srch_children_count, srch_booking_window, and srch_room_count as moderately important variables; prop_log_historical_price, orig_destination_distance, and comp1

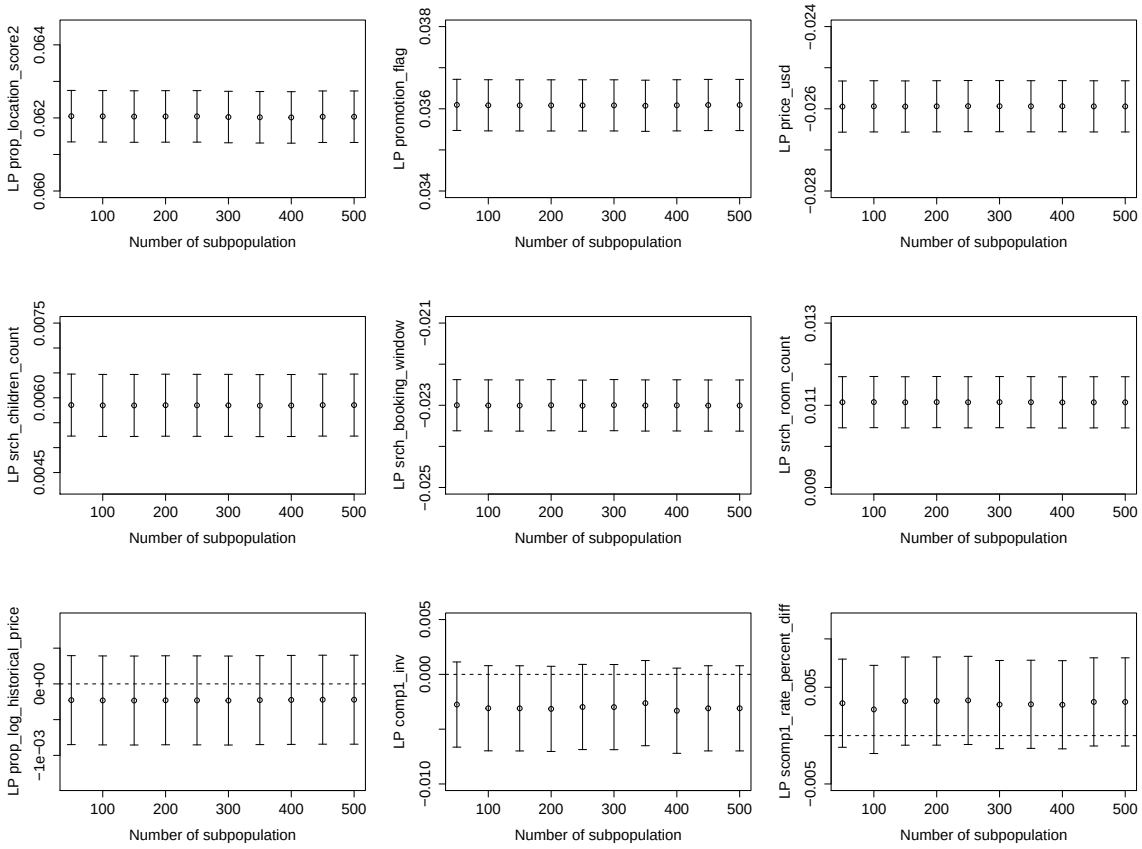

Figure 9: LP statistics and 95% confidence intervals for nine variables across different numbers of subpopulations (dotted line is at zero).

`_rate_percent_diff`, as three less influential variables, and compute their combined LP-statistics and 95% confidence intervals based on 10 different random partition schemes with different numbers and sizes of subpopulations. Figure 9 suggests that combined LP statistic estimates do not change dramatically as number of subpopulations increase, evidence of stable estimation.

5. Understanding Simpson's and Stein's Paradox from MetaLP Perspective. Heterogeneity is not solely a big data phenomenon, it can easily arise in small data setup. We will show in this section two smoking-gun examples– Simpson's Paradox and Stein's Paradox– where blind aggregation *without paying attention to the underlying inherent heterogeneity* leads to a misleading conclusion.

5.1. *Simpson's Paradox.* Table 3 shows the UC Berkeley admission rates (Bickel et. al. 1975) by department and gender. Looking only at the university level admission rates at the bottom of this table, there appears to be a significant different in admission rates for males at 45% and females at 30%. However, the department level data *does not* appear to support a strong gender bias as in the university level data. The real question at hand is whether *there*

Dept	Male	Female
A	62% (512 / 825)	**82%** (89 / 108)
B	63% (353 / 560)	**68%** (17 / 25)
C	**37%** (120 / 325)	34% (202 / 593)
D	33% (138 / 417)	**35%** (131 / 375)
E	**28%** (53 / 191)	24% (94 / 393)
F	**6%** (22 / 373)	7% (24 / 341)
All	**45%** (1198 / 2691)	30% (557 / 1835)

TABLE 3
UC Berkeley admission rates by gender by department (Bickel 1975)

is a gender bias in university admissions? We provide a concrete statistical solution to the question put forward by Pearl (2014) regarding the validity and applicability of traditional statistical tools in answering the real puzzle of Simpson's Paradox: "So in what sense do B-K plots, or ellipsoids, or vectors display, or regressions etc. contribute to the puzzle? They don't. They can't. Why bring them up? Would anyone address the real puzzle? It is a puzzle that cannot be resolved in the language of traditional statistics."

In particular, we will demonstrate how adopting the `MetaLP` modeling and combining strategy (that properly takes the existing heterogeneity into account) can resolve issues pertaining to Simpson's paradox (Simpson 1951). This simple example teaches us that *simply averaging* as a means of combining effect-size is *not appropriate* irrespective of small or big data. The calculation for weights *must* take into account the underlying departure from homogeneity, which is ensured in the `MetaLP` distributed inference mechanism. Now we explain how this paradoxical reversal can be resolved using the `MetaLP` technology.

As both admission (Y) and gender (X) are binary variables, we can compute at most one LP-orthogonal polynomial for each variable $T_1(Y;Y)$ and $T_1(X;X)$; accordingly we can compute only the first-order linear LP statistics $\mathrm{LP}[1;Y,X]$ for each department. Following Equation (3.9), we derive and estimate the aCD for the LP statistic for each of the 6 departments, $H(\mathrm{LP}_l[1;X,Y])$, $l=1,\ldots,6$, and for the aggregated university level dataset, $H(\mathrm{LP}_a[1;X,Y])$. As noted in Section 3.3, the department level aCDs are normally distributed with a mean of $\widehat{\mathrm{LP}}_l[1;X,Y]$ and variance of $1/n_\ell$ where n_ℓ is the number of applicants to department l. Similarly the aggregated aCD is also normally distributed with a mean of $\widehat{\mathrm{LP}}_a[1;X,Y]$ and variance of $1/n_a$ where n_a is the number of applicants across all departments.

Apply heterogeneity-corrected `MetaLP` algorithm following Theorem 3.5 to estimate the combined aCD across all departments as follows:

$$H^{(c)}(\mathrm{LP}[1;X,Y]) = \Phi\left[\left(\sum_{\ell=1}^{6}\frac{1}{\tau^2+(1/n_\ell)}\right)^{1/2}(\mathrm{LP}[1;X,Y]-\widehat{\mathrm{LP}}^{(c)}[1;X,Y])\right] \quad \text{with}$$

$$\widehat{\mathrm{LP}}^{(c)}[1;X,Y]) = \frac{\sum_{\ell=1}^{6}(\tau^2+(1/n_\ell))^{-1}\widehat{\mathrm{LP}}_\ell[1;X,Y])}{\sum_{\ell=1}^{6}(\tau^2+(1/n_\ell))^{-1}}$$

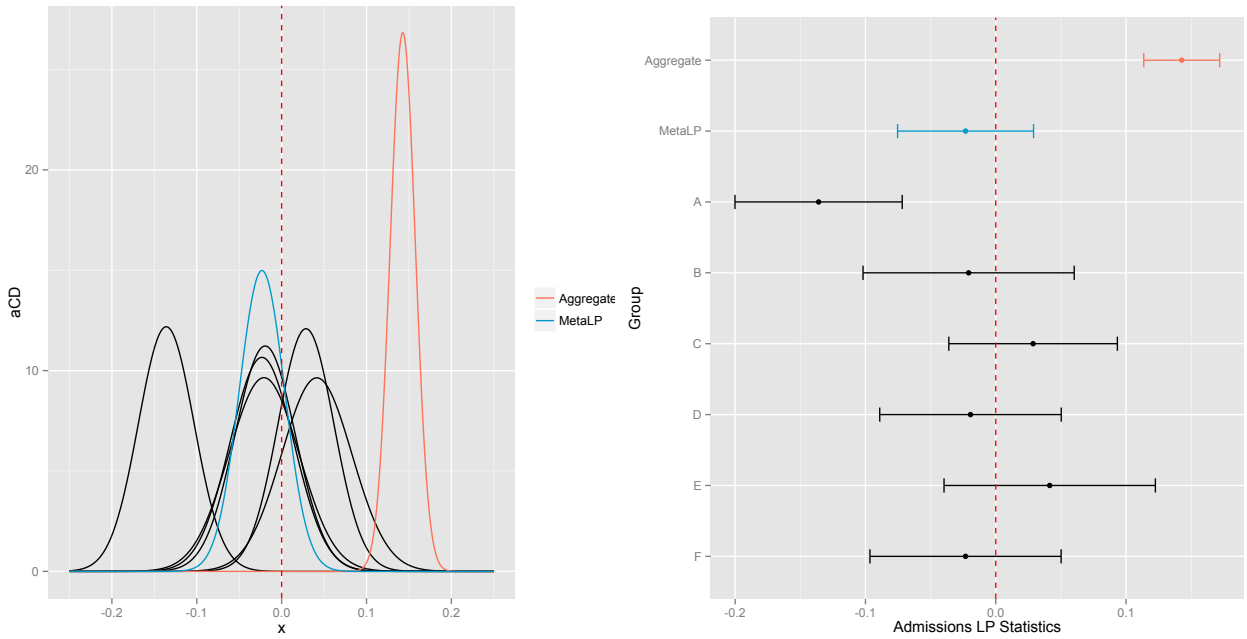

Figure 10: (color online) (a) Left: aCDs for linear LP statistics for UC Berkeley admission rates by gender (department level aCDs in black); (b) Right: 95% Confidence Intervals for LP statistics for UC Berkeley admission rates by gender.

where $\widehat{\mathrm{LP}}^{(c)}[1;X,Y])$ and $\sum_{l=1}^{6}(\tau^2+(1/n_\ell))^{-1}$ are the mean and variance respectively of the meta-combined aCD for $\mathrm{LP}[1;X,Y]$. Here, the heterogeneity parameter τ^2 is estimated using the restricted maximum likelihood formulation outlined in Supplementary Section D. Figure 10(a) displays the estimated aCDs for each department, aggregated data, and for the `MetaLP` method. First note that the aggregated data aCD is very different from the department level aCDs, which is characteristic of the Simpson's paradox reversal phenomenon due to naive "aggregation bias". This is why the aggregated data inference suggests a gender bias in admissions while the department level data does not. Second note that the aCD from the `MetaLP` method provides an estimate that falls more in line with the department level aCDs. This highlights the advantage of the `MetaLP` meta-analysis framework for combining information in a judicious manner. Also, as mentioned in Section 3.3, all traditional forms of statistical inference (e.g. point and interval estimation, hypothesis testing) can be derived from the aCD above.

For example, we can test $H_0 : \mathrm{LP}^{(c)}[1;X,Y] \leq 0$ (indicating no male preference in admissions) vs. $H_1 : \mathrm{LP}^{(c)}[1;X,Y] > 0$ (indicating a male preference in admissions) using the aCD for $\mathrm{LP}^{(c)}[1;X,Y]$. The corresponding p-value for the test comes from the probability associated with the support of H_0, $C = (-\infty, 0]$, (i.e. "high" support value for H_0 leads to acceptance) following Xie and Singh (2013). Hence the p-value for the above test becomes

$$\text{p-value} = H\big(0;\mathrm{LP}^{(c)}[1;X,Y]\big) = \Phi\left(\frac{0-\widehat{\mathrm{LP}}^{(c)}[1;X,Y]}{\sqrt{\sum_{l=1}^{6}(\tau^2+(1/n_l))^{-1}}}\right) \approx .81.$$

In this case the support of the LP-CD inference (also known as 'belief' in fiducial literature, Kendall and Stuart, 1974) is .81. Hence at the 5% level of significance, we strongly accept H_0 and confirm that there is no evidence to support a significant gender bias favoring males in admissions using the `MetaLP` approach.

In addition we can also compute the 95% confidence intervals for the LP statistics measuring the significance of the relationship between gender and admissions as shown in Figure 10(b). Note (the paradoxical reversal) 5 out of the 6 departments show no significant gender bias at the 5% level of significance as the confidence intervals include positive and negative values, while the confidence interval for the aggregated dataset indicates a significantly higher admission rate for males. On the other hand note that the `MetaLP` approach resolves the paradox (which arises *due to the failure of recognizing* the presence of heterogeneity among the department-based admission patterns) and correctly concludes that no significant gender bias exists as the confidence interval for the `MetaLP`-driven LP statistic includes the null value 0.

5.2. *Stein's Paradox.* Perhaps the most popular and classic dataset of Stein's paradox is given in Table 4, which shows the batting averages of 18 major league players through their first 45 official at-bats of the 1970 season. The goal is to predict each player's batting average over the remainder of the season (comprising about 370 more at bats each) using only the data of the first 45 at-bats.

Stein's shrinkage estimator (James and Stein, 1961), which can be interpreted as an empirical Bayes estimator (Efron and Morris, 1975) turns out to be more than 3 times as efficient than the MLE estimator. Here we provide a `MetaLP` approach to this problem by recognizing the "parallel" structure (18 parallel sub-populations) of baseball data, which fits nicely into the "decentralized" `MetaLP` modeling framework.

We start by defining the variance-stabilized effect-size estimates for each group

$$\widehat{\theta}_i = \sin^{-1}(2\hat{\mu}_i^{(\mathrm{MLE})} - 1), \quad i = 1, \ldots, k$$

whose asymptotic distribution is normal with mean θ_i and variance $1/n_i$ where $n_i = 45$ (for all i) is the number of at-bats for each player and $\hat{\mu}_i^{(\mathrm{MLE})}$ is the individual batting average for player i. Figure 11 provides some visual evidence of the heterogeneity between the studies.

We apply a MetaLP procedure that incorporates inter-study variations and is applicable for unequal variance/sample size scenarios with no further adjustment. First we estimate the weighted mean, $\hat{\theta}_\mu$, of the transformed batting averages with weights for each study $(\hat{\tau}^2_{\mathrm{DL}} + n_i^{-1})^{-1}$, where $\hat{\tau}^2_{\mathrm{DL}}$ denotes the DerSimonian and Laird data-driven estimate given in Appendix D. The `MetaLP` estimators, $\hat{\theta}_i^{(\mathrm{LP})}$, are represented as weighted average between the transformed batting averages and $\hat{\theta}_\mu$ as follows:

$$\hat{\theta}_i^{(\mathrm{LP})} = \lambda\hat{\theta}_\mu + (1-\lambda)\widehat{\theta}_i, \qquad (i = 1, \ldots, 18),$$

where $\lambda = (n_i^{-1})/(\hat{\tau}^2_{\mathrm{DL}} + n_i^{-1})$. Table 4 shows that MetaLP-based estimators are as good as James–Stein empirical Bayes estimators for the baseball data. This stems from the simple fact that random-effect meta-analysis and the Stein estimation are mathematically equivalent. But nevertheless, the framework of understanding and interpretations are different. Additionally, MetaLP is much more flexible and automatic in the sense that it works for 'any' estimators (such as mean, regression function, classification probability) beyond mean and Gaussianity

Name	hits/AB	$\hat{\mu}_i^{(\mathrm{MLE})}$	μ_i	$\hat{\mu}_i^{(JS)}$	$\hat{\mu}_i^{(\mathrm{LP})}$
Clemente	18/45	.400	.346	**.294**	.276
F Robinson	17/45	.378	.298	**.289**	.274
F Howard	16/45	.356	.276	.285	**.272**
Johnstone	15/45	.333	.222	.280	**.270**
Berry	14/45	.311	.273	**.275**	.268
Spencer	14/45	.311	.270	.275	**.268**
Kessinger	13/45	.289	.263	.270	**.265**
L Alvarado	12/45	.267	.210	.266	**.263**
Santo	11/45	.244	.269	**.261**	**.261**
Swoboda	11/45	.244	.230	**.261**	**.261**
Unser	10/45	.222	.264	.256	**.258**
Williams	10/45	.222	.256	**.256**	.258
Scott	10/45	.222	.303	.256	**.258**
Petrocelli	10/45	.222	.264	.256	**.258**
E Rodriguez	10/45	.222	.226	**.256**	.258
Campaneris	9/45	.200	.286	.252	**.256**
Munson	8/45	.178	.316	.247	**.253**
Alvis	7/45	.156	.200	**.242**	.251

TABLE 4

Batting averages $\hat{\mu}_i^{(\mathrm{MLE})}$ for 18 major league players early in the 1970 season; μ_i values are averages over the remainder of the season. The James-Stein estimates $\hat{\mu}_i^{(JS)}$ and MetaLP estimates $\hat{\mu}_i^{(\mathrm{LP})}$ provide much more accurate overall predictions for the μ_i values compared to MLE. MSE ratio for $\hat{\mu}_i^{(JS)}$ to $\hat{\mu}_i^{(\mathrm{MLE})}$ is 0.283 *and MSE ratio for $\hat{\mu}_i^{(\mathrm{LP})}$ to $\hat{\mu}_i^{(\mathrm{MLE})}$ is* 0.293 *showing comparable efficiency.*

assumptions. We feel the MetaLP viewpoint is also less mysterious and clearly highlights the core issue of heterogeneity. Our analysis indicates an exciting frontier of future research at the interface of MetaLP, Empirical Bayes, and Stein's Paradox to develop new theory of distributed massive data modeling.

6. Final Remarks on Big Data Statistical Inference. To address methodological and computational challenges for big data analysis, we have outlined a general theoretical foundation in this article, which we believe may provide the missing link between small data and big data science. Our research shows how the traditional and modern 'small' data modeling tools can be successfully adapted and judiciously connected for developing powerful big data analytic tools by leveraging state-of-the-art distributed computing environments.

In particular, we have proposed a nonparametric two sample inference algorithm that has the following two-fold practical significance for solving real-world data mining problems: (1) scalability for large data by exploiting the distributed computing architectures using a confidence distribution based meta-analysis framework, and (2) automation for mixed data by using a united LP computing formula. Undoubtedly our theory can be adapted for other common data mining problems, and we are currently investigating how the proposed framework can be utilized to develop parallelizable regression and classification algorithms for big data.

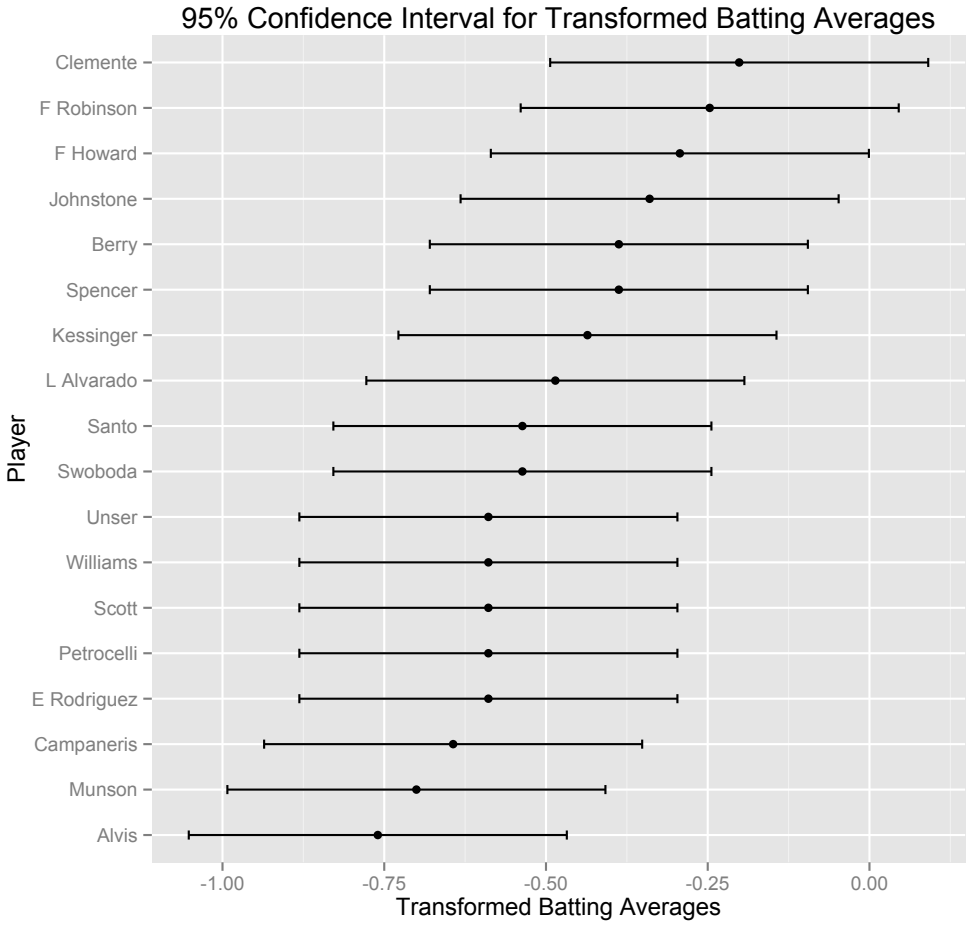

Figure 11: 95% confidence intervals for transformed batting averages, θ_i, for each player. This plot clearly indicates the heterogeneity of the effect sizes estimates.

Instead of developing distributed versions of statistical algorithms on a case-by-case basis, here we develop a systematic and automatic strategy that will provide a generic platform to extend traditional and modern statistical modeling tools to large datasets using scalable distributed algorithms, thus addressing one of the biggest bottlenecks for data-intensive statistical inference. We believe this research is a great stepping stone towards developing a United Statistical Algorithm (Parzen and Mukhopadhyay, 2013) to bridge the increasing gap between the theory and practice of small and big data analysis.

Acknowledgements. SM thanks William S. Cleveland for pointing out relevant literature and for many helpful comments. This research is partially supported by Fox 2014 young scholars interdisciplinary grant and by the Fox School PhD student research competition award. We would also like to thank the editor, associate editor and three anonymous referees whose comments and suggestions considerably improved the presentation of the paper.

References.

[1] Bickel, P. J., Hammel, E. A., and O'Connell, J. W. (1975). Sex bias in graduate admissions: Data from Berkeley. *Science* **187**, 398–403.

[2] Chen, X. and Xie, M.-g. (2014). A split-and-conquer approach for analysis of extraordinarily large data. Statist. Sinica, 24(4):1655-1684.

[3] DerSimonian R, Laird N. (1986) Meta-analysis in clinical trials. *Controlled Clinical Trials.* **7**, 177–188.

[4] Efron, B. and Morris, C. (1975). Data analysis using stein's estimator and its generalizations. *Journal of the American Statistical Association*, **70**, 311–319.

[5] Guha, S., Hafen, R., Rounds, J., Xia, J., Li, J., Xi, B., and Cleveland, W. S. Large complex data: divide and recombine (d&r) with rhipe. *Stat* 1.1 (2012): 53-67.

[6] Hedges, Larry V., and Ingram, Olkin. (1985). Statistical method for meta-analysis, London: Academic Press.

[7] Higgins, Julian P.T., Thompson, Simon G. (2002). Quantifying heterogeneity in a meta-analysis. *Statistics in Medicine*. 21:1539–1558 (DOI: 10.1002/sim.1186)

[8] Hunter, J. E. and F. L. Schmidt (2004). *Methods of meta-analysis: Correcting error and bias in research findings*. Sage.

[9] James, W. and Stein, C. (1961). Estimation with quadratic loss. *Proceedings of the fourth Berkeley symposium on mathematical statistics and probability*, **1** 361–379.

[10] Kaggle Inc. (2013) Personalize Expedia Hotel Searches - ICDM 2013. *Available online at:* https://www.kaggle.com/c/expedia-personalized-sort/data

[11] Kleiner, A., Talwalkar, A., Sarkar, P., and Jordan, M. I. (2014). A scalable bootstrap for massive data. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 76(4):795-816.

[12] Liang, F., Cheng, Y., Song, Q., Park, J., and Yang, P. (2013). A resampling-based stochastic approximation method for analysis of large geostatistical data. J. Amer. Statist. Assoc., 108(501):325- 339.

[13] Lin, N. and Xi, R. (2011). Aggregated estimating equation estimation. Stat. Interface, 4(1):73-83.

[14] Liu, L. (2012). Computing infrastructure for big data processing. *Frontiers of Computer Science* **7**, 165–170.

[15] Mukhopadhyay, S. and Parzen, E. (2014). LP approach to statistical modeling. *arXiv:1405.2601*.

[16] Parzen, E. (2013). Discussion of "Confidence distribution, the frequentist distribution estimator of a parameter: a review" by Min-ge Xie and Kesar Singh. *International Statistical Review 81*(1), 48–52.

[17] Parzen, E. and S. Mukhopadhyay (2013). LP Mixed Data Science: Outline of Theory. *arXiv:1311.0562*.

[18] Parzen, E. and Mukhopadhyay, S. (2013). United Statistical Algorithms, Small and Big Data, Future of Statisticians. *arXiv:1308.0641*.

[19] Pearl, J. (2014). Comment: Understanding simpson's paradox using a graph. *Available online at: http://andrewgelman.com/2014/04/08/understanding-simpsons-paradox-using-graph/*.

[20] Schweder, T., and Hjort, N. L. (2002). Confidence and Likelihood. *Scandinavian Journal of Statistics.* **29**, 309–332.

[21] Simpson, E. H. (1951). The interpretation of interaction in contingency tables. *Journal of the Royal Statistical Society. Series B* **13**(2), 238–241.

[22] Singh, K., Xie, M. and Strawderman, W.(2005). Combining Information from Independent Sources through Confidence Distributions. *The Annals of Statistics* **1**, 159-183.

[23] Sutton, A.J., and Higgins, J.P. (2008). Recent developments in meta-analysis. *Statistics in Medicine* **27**, 625-650.

[24] Xie, M., Singh, K., and W. E. Strawderman (2011). Confidence distributions and a unifying framework for meta-analysis. *Journal of the American Statistical Association* **106**, 320–333.

[25] Xie, M. and Singh, K.(2013). Confidence Distribution, the Frequentist Distribution Estimator of a Parameter: A Review. *International Statistical Review,* **81**, 48–52.

[26] Wiener, N. (1938). The homogeneous chaos. *American Journal of Mathematics* **60** 897–936.

SUPPLEMENTARY MATERIAL

Section A provides simulation studies to further investigate the performance (in terms of both statistical accuracy and computational efficiency) of our proposed MetaLP distributed learning scheme. Also, it is reasonable to ask the question whether or not the `MetaLP`-inspired algorithms are applicable for small-data in the sense that it can closely approximate the "oracle" full data-based inference solution. Section B will shed light on this aspect. Finally, Sections C and D will describe a `MapReduce` computational implementation of the MetaLP inference engine.

A. Simulation Studies. We investigate the statistical efficiency from two perspectives: first, we test if LP combined estimators from our `MetaLP` method are consistent with oracle full data LP estimators, which reflects the validity of our computational learning model from a statistical estimation standpoint; second, we investigate if our method is able to correctly identify important variables and noisy variables – the statistical inference part.

In our simulation settings, the dataset has the form $(\mathbf{X}_i, Y_i) \sim P$, i.i.d, for $i = 1, 2, ..., n$, where $\mathbf{X}_i \in \mathbb{R}^p$ and $Y_i \in (0, 1)$. We generate dataset from model $Y_i \sim \text{Bernoulli}(P(\beta_1 X_{1i}^2 + \mathbf{X}_{(-1)i}^T \boldsymbol{\beta}_{-1}))$, where $P(u) = \exp(u)/(1 + \exp(u))$. $\mathbf{X}_{-1}$ and $\boldsymbol{\beta}_{-1}$ mean all X's except X_1 and all β's except β_1. We set $\boldsymbol{\beta} = (\beta_1, \beta_2, ..., \beta_p) = (3, -2, 1.5, 0, ..., 0)^T$ to be a p-dimensional coefficient vector, where $p = 50$, and then generate X_{1i}, X_{2i}, and X_{3i} from StudentT(30), Poisson(2), and Bernoulli($p = 0.4$) respectively. The remaining features are all generated from the standard Normal distribution. Let n be the total number of observations in one dataset, where $n = 5,000, 50,000, 500,000,$ and 1 million. For each setting of n, we generate 50 datasets and randomly partition the dataset with n total observations into $k = \lfloor n^\gamma + 0.5 \rfloor$ subpopulations, where $\gamma = 0.3, 0.4, 0.5$. For each partitioning scheme, we compare the LP statistic estimation errors (mean absolute error) across all p variables and the resulting inference from the full data approach and the distributed MetaLP approach.

Figure 12 shows the results for mean absolute errors of first-order LP statistics across all p variables for the simulated setting. All mean absolute errors are small, indicating the estimation using the distributed `MetaLP` approach is consistent with estimation using the whole dataset. Moreover, for every n, as the number of partitions k increase, errors increase correspondingly; for fixed k, errors are inversely proportional to the number of observations n. This indicates that partitioning the dataset into too many subpopulations may influence the final results, but not significantly as all errors are still extremely small. The results also give data scientists practical ideas on how to partition the dataset appropriately.

For 50 repetitions, we apply both the `MetaLP` and full data based LP variable selection methods and monitor the accuracy and computation time. Second order LP statistics are used to test for significance for X_1 since it has a second order impact on the dependent variable, and first order LP statistics are used to detect significance of other variables. Table 5 reports the accuracy of both `MetaLP` and the full data based LP variable selection methods in including the true model: $\{X_1, X_2, X_3\}$ along with run times. Note that both methods correctly select all the three important variables every time, which suggests that the distributed approach is comparable to the full data approach in selecting important variables. However, our distributed framework `MetaLP` saves a considerable amount of time compared to the non-distributed computing approach (i.e. computing LP statistics from the whole dataset). We

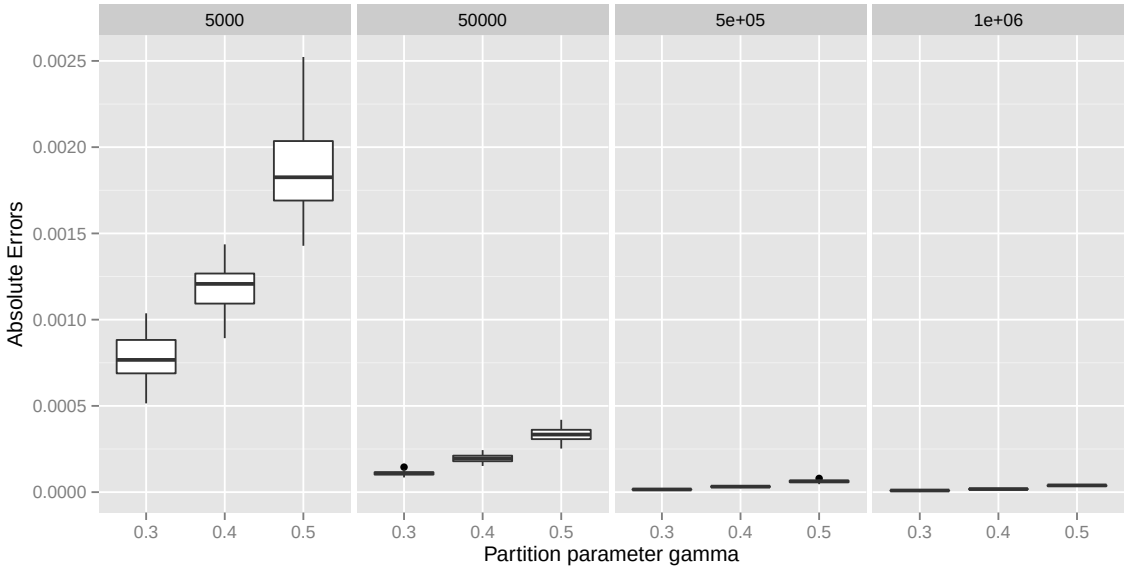

Figure 12: The absolute error between distributed MetaLP and full-data LP estimates under different simulation settings. As the size of the data set increases our methods performs better in the sense of lower estimation error.

list speed improvement (how many times faster the `MetaLP` is over the full data approach) in the last column of Table 5. For example, when $n = 1,000,000$ and $\gamma = 0.5$, `MetaLP` is about *150 fold faster than full data based* LP *method*.

B. MetaLP Analysis of Titanic Data. The *Titanic* dataset is utilized as a benchmark to validate its effectiveness, accuracy, and robustness. Due to its manageable size, we are able to compute the aggregated data-based (with no partitioning) LP-estimate (oracle answer) and can compare with the MetaLP answer, which operates under a distributed computing framework. A step-by-step MetaLP analysis of Titanic dataset is given below.

The *Titanic* dataset contains information on 891 of its passengers, including which passengers survived. A key objective in analyzing this dataset is to better understand which factors (e.g. age, gender, class, etc.) significantly influence passenger survival. Complete descriptions of all 8 variables can be found in Table 6. We seek to estimate the relationship between various passenger characteristics $(X_i, i = 1, \ldots, 7)$ and the binary response variable (Y), passenger survival, by using both our distributed algorithm and traditional aggregated LP statistics to compare their results.

To develop an automatic solution to the mixed data problem we start by constructing LP-score polynomials for each variable. Figure 13 shows the shapes of LP-bases for two variables from the Titanic data. Next we randomly assign 891 observations to 5 different subpopulations and calculate LP statistics for each variable in each subpopulation, and then combine LP statistics to get a combined LP statistic for each variable. We repeat this process three times to see how much our final MetaLP result changes with different random partitions of the full data. Figures 14(a) shows the I^2 statistics for three random partitions on the *Titanic* dataset.

Observations n	Partition Parameter γ	Accuracy		Run Time (seconds)		Speed Increase
		Full Data	`MetaLP`	Full Data	`MetaLP`	
5,000	0.3		1		0.09	10.9
	0.4	1	1	0.98	0.08	12.3
	0.5		1		0.05	19.6
50,000	0.3		1		0.38	20.9
	0.4	1	1	7.95	0.20	39.8
	0.5		1		0.13	61.2
500,000	0.3		1		2.02	41.4
	0.4	1	1	83.66	1.01	82.8
	0.5		1		0.67	124.9
1,000,000	0.3		1		3.36	53.2
	0.4	1	1	178.65	1.59	112.4
	0.5		1		1.17	152.7

TABLE 5
Accuracy and computational run time of parallelized `MetaLP` and full-data based LP *method in including the true model* $\{X_1, X_2, X_3\}$.

Variable Name	Type	Description	Value
Survival	Binary	Survival	0 = No; 1 = Yes
Pclass	Categorical	Passenger Class	1 = 1st; 2 = 2nd; 3 = 3rd
Sex	Binary	Sex	Male; Female
Age	Continuous	Age	0 - 80
Sibsp	Discrete	Number of Siblings/Spouses Aboard	0 - 8
Parch	Discrete	Number of Parents/Children Aboard	0 - 6
Fare	Continuous	Passenger Fare	0 - 512.3292
Embarked	Categorical	Port of Embarkation	C = Cherbourg; Q = Queenstown; S = Southampton

TABLE 6
Data dictionary for the Titanic dataset. Small yet mixed-data problem.

Even with the randomly assigned partitions, some variables may exhibit heterogeneity among subpopulations as I^2 statistics above 40%, e.g. random partition 2 results show heterogeneity in variables Embarked and Sex. Thus, we use τ^2 regularization to handle the problem. Figure 14(b) shows the I^2 statistics after τ^2 regularization. The additional τ^2 parameter accounts for the heterogeneity in the subpopulations and adjusts the estimators accordingly, resulting in significantly lower I^2 statistics for all variables under this model.

Figure 15 contains the LP statistics and their 95% confidence intervals generated from our algorithm for 3 repetitions of random groupings ($k = 5$) along with the confidence intervals generated using the whole dataset. A *remarkable result of our method* is that the MetaLP estimators and the aggregated (based on entire data) LP-estimators (both point and interval estimators) are almost indistinguishable for *all* the variables. In summary, the estimators from our MetaLP method produces very similar inference to the estimators using the entire dataset, which means we can obtain accurate and robust statistical inference while taking advantage of the computational efficiency in parallel, distributed processing.

C. MapReduce Computation Framework and R Functions. In this note we describe how the proposed `MetaLP` statistical algorithmic framework for big data analysis can

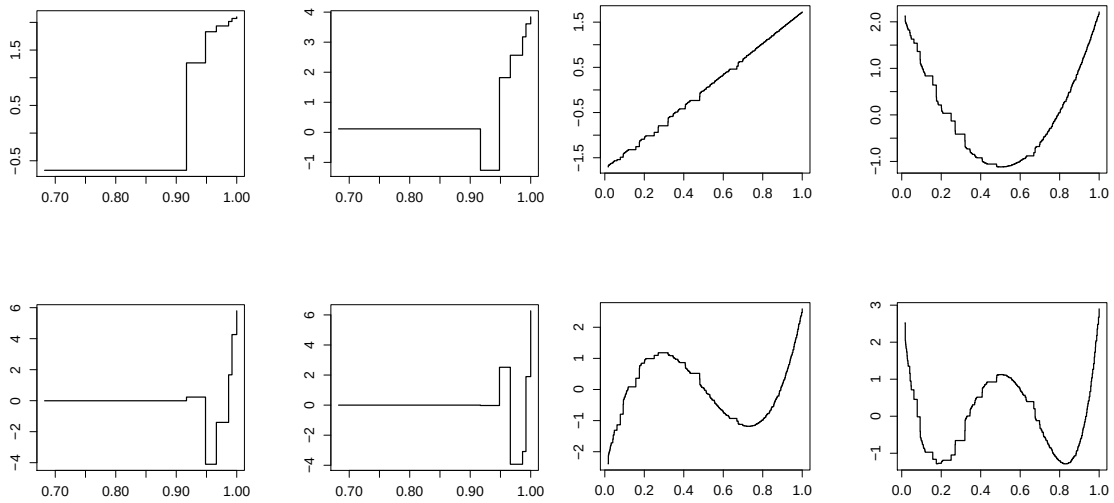

Figure 13: (a) Left panel shows the shape of the first four LP orthonormal score functions for the variable # Siblings/Spouses Aboard, which is a discrete random variable takes values $0, \ldots, 8$; (b) Right: the shape of the LP basis for the continuous variable Passenger Fare.

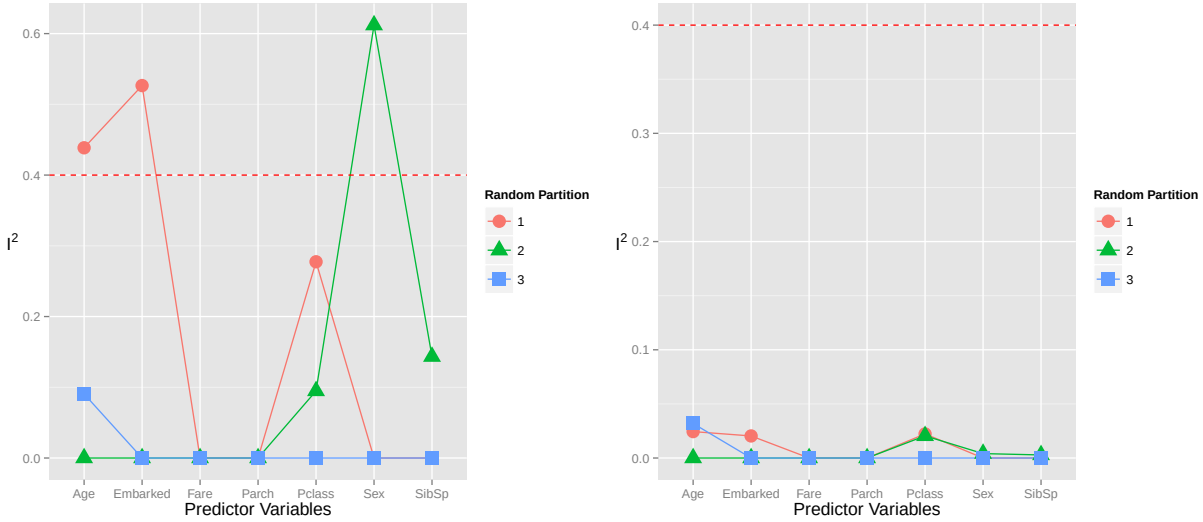

Figure 14: (color online) (a) Left: I^2 diagnostic of the meta-analysis inference by Theorem 3.1 for three random partitions on the *Titanic* dataset (b) Right: I^2 diagnostic with τ^2 regularization on the *Titanic* dataset for three random partitions.

easily be integrated with the `MapReduce` computational framework, along with the required `R` code. MapReduce implementation of `MetaLP` allows efficient parallel processing of large amounts of data to achieve scalability.

C1. LP.Mapper. We apply the following `LP.Mapper` function to each subpopulation. This function computes $\mathrm{LP}[j; X, Y]$ for $j = 1, \ldots, m$ (where user selects m, which should be less than the number of distinct values of the given random sample). The first step is to design

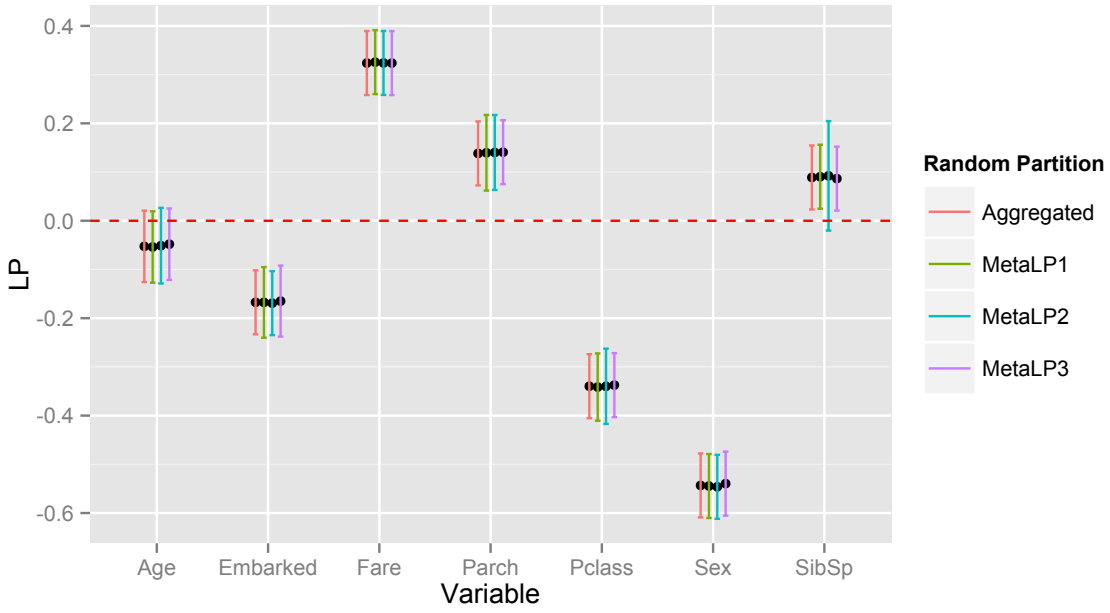

Figure 15: (color online) 95% Confidence Interval of LP Statistic for each variable based on three `MetaLP` repetitions and aggregated full dataset (which is the oracle estimate).

the data-adaptive orthogonal LP polynomial transformation of the given random variable X. This is implemented using the function `LP.Score.fun`. The second step uses the LP inner product to calculate the LP variable selection statistic using the function `LP.VarStat` (see Section 3.1 for details).

Inputs of `LP.Mapper`. Y is binary (or discrete multinomial) and X is a mixed (discrete or continuous type) predictor variable.

Outputs of `LP.Mapper`. It returns the estimated $\widehat{\mathrm{LP}}[j; X, Y]$ and the corresponding (asymptotic) sample variance. Note that the sample LP statistic converges to $\mathcal{N}(0, \sigma_\ell^2 = 1/n_\ell)$, where n_ℓ is the effective sample size of the ℓth subpopulation. By effective size we mean $n_\ell - M_\ell(X)$, where $M_\ell(X)$ denotes the number of missing observations for variable X in the ℓth partition. `LP.Mapper` returns only $\{\widehat{\mathrm{LP}}[1; X, Y], \ldots, \widehat{\mathrm{LP}}[m; X, Y]\}$ and n_ℓ, from which we can easily reconstruct the CD of the LP statistics.

```
LP.Mapper <- function (Y,x,m=1) {
  LP.Score.fun <- function(x,m){
  # ------>this function is called by the main LP.VarStat()
    u <- (rank(x,ties.method = c("average")) - .5)/length(x)
    m <- min(length(unique(u ))-1, m  )
    S.mat <- as.matrix(poly(u ,df=m))
    return(as.matrix(scale(S.mat)))
  }
  LP.VarStat <- function(Y,x,m){
```

```
#--> accept variable x and Y 0/1 returns LP variable selction stat
  x <- ifelse (x=="NULL",NA,x)
  x <- na.omit (x) ### eliminate values of NA
  if (length (unique(x)) <=1 ) { #--> if all x are the same then r.lp=0
    r.lp=0;n=0
  } else{
    which <- na.action (x)
    if (length (which) > 0) Y <- Y[-which]
    if (length (unique(Y)) <=1) { #--> if all Y are the same then r.lp=0
      r.lp=0;n=0
    } else {
      x <- as.numeric (x)
      S <- LP.Score.fun(x,m)
      r.lp <- cor(Y,S);n=length(Y)
    }
  }
  return(c(r.lp,n))
}
  temp = LP.VarStat  (Y,x,m)
  output.LP = temp [1:length(temp)-1]
  output.n = temp [length(temp)]
  logic = ifelse (length(temp)-1==m,"NA",
                  "m is not less than the number of distinct value of x")
return (list(LP=output.LP,n=output.n,Warning=logic)) }
```

C2. Meta.Reducer. `LP.Mapper` computes the sample LP statistics and the corresponding sample variance. Now at the 'Reduce' step our goal is to judiciously combine these estimates from k subpopulations to produce the statistical inference for the original large data. Here we implement the MetaReduce strategy to combine the inference from all the subpopulations, implemented in the function `Meta.Reducer`.

Before performing the `Meta.Reducer` step we run the 'combiner' operation that gathers the outputs of the `LP.Mapper` function for all the subpopulations and organizes them in the form of a list, which has two components: (i) a matrix `L.value` of order $k \times p$, where k is the number of subpopulations and p is the number of predictor variables (the (ℓ, i)th element of that matrix stores the jth LP-Fourier statistic $\mathrm{LP}[j; X_i, Y]$ for ℓth partition); (ii) a matrix `P.size` of size $k \times p$ ((ℓ, i)th element stores the effective size of the subpopulation for the variable ℓ).

Inputs of `Meta.Reducer`

1. `L.value` and `P.size`
2. `fix`: a binary argument (TRUE or FALSE), indicating whether to ignore the τ^2 regularization. If it equals to FALSE, then the model with τ^2 regularization is applied.
3. `method`: It's valid only if `fix` equals FALSE, and can equal to either `"DL"` or `"REML"`, indicating the estimation method of τ^2.
4. `"DL"` stands for the method proposed by DerSimonian and Laird (1986), and `"REML"` is the restricted maximum likelihood method, which was proposed by Normand (1999).

We include the calculation methods of these two τ^2 in the next section.

Outputs of `Meta.Reducer`

1. Meta-analysis combined LP statistic estimators
2. Standard errors of meta-analysis combined LP statistic estimators
3. I^2 heterogeneity diagnostic
4. τ^2 estimate only if `fix` equals to FALSE

```
Meta.Reducer <- function (L.value, P.size, fix, method) {
  th_c <- NA
  sd_th_c <- NA
  for (i in 1:ncol(L.value)) {
    th_c[i] <- sum(L.value[,i]*P.size[,i])/sum(P.size[,i])
    sd_th_c[i] <- sqrt(1/sum(P.size[,i]))
  }
  Q = matrix (,ncol(L.value),1)
  for (i in 1:ncol(L.value)) {
    Q [i,] = sum ( P.size[,i]*(L.value [,i] - th_c[i])^2)
  }
  K=NA
  for (i in 1:ncol(L.value)) {
    A = P.size[,i]
    K [i] = length (A[A!=0])
  }
  if (fix==T) {
    I_sq.f <- ifelse  ((Q -(K-1))/Q>0, (Q -(K-1))/Q,0)
    return (list (LP.c=th_c, SE.LP.c=sd_th_c,I_sq.f=I_sq.f))
  } else{
    if (method=="DL"){
      tau.sq =NA
      for (i in 1:ncol(L.value)) {
        tau.sq[i] = (Q[i]-(K[i]-1)) /
          (sum(P.size[,i]) - sum((P.size[,i])^2) / sum(P.size[,i]))
      }
      tau.sq = ifelse (tau.sq>0,tau.sq,0)
      w_i = matrix (,nrow(P.size), ncol(P.size))
      for (i in 1:ncol(L.value)) {
        w_i[,i] = (1/P.size[,i]+tau.sq[i])^-1
      }
      mu.hat <- NA
      SE_mu.hat <- NA
      for (i in 1:ncol(L.value)) {
        mu.hat[i] <- sum(L.value[,i]*w_i[,i])/sum(w_i[,i])
        SE_mu.hat[i] <- sqrt(1/sum(w_i[,i]))
      }
      lam_i = matrix (,nrow(P.size), ncol(P.size))
      for (i in 1:ncol(L.value)) {
```

```
    lam_i[,i] = (1/P.size[,i])/(1/P.size[,i]+tau.sq[i])
  }
  th.tilde = matrix (,nrow(L.value), ncol(L.value))
  for (i in 1:ncol(L.value)) {
    th.tilde [,i] = lam_i[,i] * mu.hat [i] + (1-lam_i[,i])*L.value[,i]
  }
  th.tilde = ifelse (is.nan(th.tilde)==T,0,th.tilde)
  Q = matrix (,ncol(L.value),1)
  for (i in 1:ncol(L.value)) {
    Q [i,] = sum ( w_i[,i]*(th.tilde [,i] - mu.hat[i])^2)
  }
  I_sq.r <- ifelse  ((Q -(K-1))/Q>0, (Q -(K-1))/Q,0)
  return (list (LP.c=mu.hat, SE.LP.c=SE_mu.hat,I_sq.r=I_sq.r,tau.sq=tau.sq))
}
if (method=="REML") {
  tau.sq =NA
  for (i in 1:ncol(L.value)) {
    tau.sq[i] = (Q[i]-(K[i]-1)) /
      (sum(P.size[,i]) - sum((P.size[,i])^2) / sum(P.size[,i]))
  }
  tau.sq = ifelse (tau.sq>0,tau.sq,0)
  for (i in 1:ncol(L.value)) {
    if (sum(P.size[,i]==0)>0) {
      n = P.size[,i] [-which (P.size[,i]==0)]
      thh = L.value[,i] [-which (P.size[,i]==0)]
    }else{
      n = P.size[,i]
      thh = L.value[,i]
    }
    nloop = 0
    absch = 1 # absolute change in tauR2 value
    while ( absch > 10^(-10) ) {
      nloop = nloop + 1
      if ( nloop > 10^5 ) {
        tau.sq[i] = NA ; stop ### not converge
      }
      else {
        tau.sq.old <- tau.sq[i] # tauR2Old
        # update thetaR, wR
        wR <- 1/(1/n + tau.sq.old)
        thetaR <- sum(wR*thh) / sum(wR)
        # update tauR
        tau.sq[i] <- sum( wR^2*(K[i]/(K[i]-1)*(thh- thetaR)^2 - 1/n) ) / sum(wR^2)
        absch <- abs(tau.sq[i] - tau.sq.old)
      }
```

```
      }
    }
    tau.sq = ifelse (tau.sq > 0, tau.sq, 0)
    w_i = matrix (,nrow(P.size), ncol(P.size))
    for (i in 1:ncol(L.value)) w_i[,i] = (1/P.size[,i]+tau.sq[i])^-1
    mu.hat <- NA
    SE_mu.hat <- NA
    for (i in 1:ncol(L.value)) {
      mu.hat[i] <- sum(L.value[,i]*w_i[,i])/sum(w_i[,i])
      SE_mu.hat[i] <- sqrt(1/sum(w_i[,i]))
    }
    lam_i = matrix (,nrow(P.size), ncol(P.size))
    for (i in 1:ncol(L.value)) {
      lam_i[,i] = (1/P.size[,i])/(1/P.size[,i]+tau.sq[i])
    }
    th.tilde = matrix (,nrow(L.value), ncol(L.value))
    for (i in 1:ncol(L.value)) {
      th.tilde [,i] = lam_i[,i] * mu.hat [i] + (1-lam_i[,i])*L.value[,i]
    }
    th.tilde = ifelse (is.nan(th.tilde)==T,0,th.tilde)
    Q = matrix (,ncol(L.value),1)
    for (i in 1:ncol(L.value)) {
      Q [i,] = sum ( w_i[,i]*(th.tilde [,i] - mu.hat[i])^2)
    }
    I_sq.r <- ifelse  ((Q -(K-1))/Q>0, (Q -(K-1))/Q,0)
    return (list (LP.c=mu.hat, SE.LP.c=SE_mu.hat,I_sq.r=I_sq.r,tau.sq=tau.sq))
  }
 }
}
```

D. τ^2 estimator. There are many different proposed estimators for the τ^2 parameter. We consider the DerSimonion and Laird estimator (DerSimonian and Laird 1986) ($\hat{\tau}^2_{\mathrm{DL}}$) and the restricted maximum likelihood estimator ($\hat{\tau}^2_{\mathrm{REML}}$) for our analysis. $\hat{\tau}^2_{\mathrm{DL}}$ can be found from the following equation:

$$\hat{\tau}^2_{DL} = \max\left\{0, \frac{Q-(k-1)}{\sum_\ell s_\ell^{-2} - \sum_\ell s_\ell^{-4}/\sum_\ell s_\ell^{-2}}\right\}; \tag{6.1}$$

where

$$Q = \sum_{\ell=1}^{k}\left(\widehat{\mathrm{LP}}_\ell[j;X,Y] - \widehat{\mathrm{LP}}^{(c)}[j;X,Y]\right)^2 s_\ell^{-2}.$$

However, $\hat{\tau}^2_{\mathrm{REML}}$ should be calculated in an iterative fashion to maximize the restricted likelihood following these steps:

Step 1: Obtain the initial value, $\hat{\tau}^2_0$. We used $\hat{\tau}^2_{\mathrm{DL}}$ as the initial value:

$$\hat{\tau}_0^2 = \hat{\tau}_{\text{DL}}^2.$$

Step 2: Obtain $\widehat{\text{LP}}_\tau^{(c)}[j; X, Y]$ (τ-corrected combined LP statistics).

$$\widehat{\text{LP}}_\tau^{(c)}[j; X, Y] = \frac{\sum_\ell w_\ell(\tau_0^2)\widehat{\text{LP}}_\ell[j; X, Y]}{\sum_\ell w_\ell(\hat{\tau}_0^2)}; \qquad w_\ell(\hat{\tau}_0^2) = (s_\ell^2 + \hat{\tau}_0^2)^{-1}.$$

Step 3: Obtain the REML estimate.

$$\hat{\tau}_{\text{REML}}^2 = \frac{\sum_\ell w_\ell^2(\hat{\tau}_0^2)\left(\frac{k}{k-1}\left(\widehat{\text{LP}}_\ell[j; X, Y] - \widehat{\text{LP}}_\tau^{(c)}[j; X, Y]\right) - s_\ell^2\right)}{\sum_\ell w_\ell^2(\hat{\tau}_0^2)}.$$

Step 4: Compute new $\widehat{\text{LP}}_\tau^{(c)}[j; X, Y]$ by plugging $\hat{\tau}_{\text{REML}}^2$ obtained in Step 3 into formula from Step 2.

Step 5: Repeat Step 2 and Step 3 until $\hat{\tau}_{\text{REML}}^2$ converges.

Convergence can be measured as the absolute difference between $\hat{\tau}_{\text{REML}}^2$ from the latest iteration and the previous iteration reaching a threshold close to zero.
().

References.

[1] Kaggle Inc. (2012) Titanic: Machine Learning from Disaster. *Available online at:* https://www.kaggle.com/c/titanic-gettingStarted/data

[2] Kendall, M. and Stuart, A. (1974). The Advanced Theory of Statistics, 2, 3rd ed. London: Griffin.

Temple University
Department of Statistics
1801 Liacouras Walk
Philadelphia, Pennsylvania, 19122, U.S.A.
E-mail: scott.bruce@temple.edu
zeda.li@temple.edu
alex.yang@temple.edu
deep@temple.edu